1 CS 9622 - Fall 2017 Home ................................................................. 11
1.1 README ................................................................. 11
1.2 Course Announcements .................................................. 12
1.2.1 Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability .................................................. 12
1.2.1.1 Fall 2017 Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability ............ 14
1.2.2 Course Overview .................................................. 15
1.2.3 CS9622 Fall 2017 Version .............................................. 15
1.2.3.1 Eighth class (Monday) 6 Nov 2017 .................. 16
1.2.3.1.1 Notes on readings for eighth class ............. 16
1.2.3.2 Eleventh class (Monday) 27 Nov 2017 ............. 17
1.2.3.3 Fifth Class Monday 16 Oct 2017 .............. 18
1.2.3.3.1 Notes on readings for fifth class .............. 19
1.2.3.3.2 RailsConf 2017 - Unreasonable Estimates and Improbable Goals by Adam Sanderson (talk) .............. 20
1.2.3.3.3 RubyConf 2015 - I Estimate this Talk will be 20 Minutes Long, Give or Take 10 Minutes (talk) ...... 20
1.2.3.4 First Class (Monday) 11 Sept 2017 ............. 21
1.2.3.5 Format for Progress Reports ............................ 21
1.2.3.6 Fourth Class (Monday, 2 Oct 2017) ............... 21
1.2.3.6.1 Notes on readings for Fourth Class (Monday, 2 Oct 2017) .................................................. 22
1.2.3.7 Marking rubric for presentations ...................... 24
1.2.3.8 Ninth class (Monday) 13 Nov 2017 ..................... 25
1.2.3.8.1 notes re readings for 13 Nov 2017 ............ 26
1.2.3.9 Second Class (Monday) 18 Sept 2017 ............ 26
1.2.3.9.1 Notes on readings for second class ............ 28
1.2.3.10 Seventh class (Monday) 30 Oct 2017 ............ 29
1.2.3.10.1 Notes for seventh class readings .......... 30
1.2.3.11 Sixth Class (Monday) 23 Oct 2017 ............ 31
1.2.3.11.1 Notes re readings for sixth class .......... 32
1.2.3.12 Tenth class (Monday) 20 Nov 2017 .......... 33
1.2.3.12.1 Notes on the readings for Tenth class (Monday) 20 Nov 2017 ........................................... 33
1.2.3.13 Third Class (Monday) 25 Sept 2017 ............ 34
1.2.3.13.1 Notes on Readings for Third Class (25 Sept 2017) .................................................. 36
1.2.3.14 Toward the final exam ........................................ 37
1.2.3.14.1 Example exam database based on material covered in first class ............. 38
1.2.3.14.2 Ruby program for processing exam database .... 40
1.2.3.14.3 Script used to build exams from database (runs the ruby program with various parameter settings and invokes pdflatex) .................. 49
1.2.3.14.4 Exam database for questions covering the first 10 classes ............. 50
1.2.3.14.5 First 10 classes, question and answers pdf ............. 63
1.2.3.14.6 First 10 classes, questions with answer key pdf ............. 63
1.2.3.14.7 First 10 classes, 50 randomly chosen questions with answer key .......... 64
1.2.3.14.8 Questions and Answers for All Eleven Classes pdf ............. 64
1.2.3.14.9 Database of Questions and answers for all eleven classes ............. 70
1.2.3.14.10 Questions with Answer Key for all eleven classes pdf ............. 70
1.2.3.14.11 Fifty randomly chosen questions from all eleven classes pdf ............. 70
1.2.4 CS9622 Fall 2018 Version of Course ......................... 79
1.2.4.1 First Class: 10 Sept 2018 .............................. 79
1.2.5 Lectures to an Empty Classroom ............................... 82
1.2.5.1 Eighth Lecture -- 9 July 2018 ......................... 82
1.2.5.1.1 Bonnie Nardi, 2017 GAD Distinguished Lecture (talk) ......................... 82
1.2.5.1.2 Computational Agroecology: Sustainable Food Ecosystem Design (paper) ............. 83
1.2.5.1.3 Computational Agroecology: Sustainable Food Ecosystem Design (talk) ............. 83
1.2.5.1.4 Design Anthropology and Co-Design - presented by Kelsey Schwenk at Real Big Things #4 (talk) ............. 83
1.2.5.1.5 Design Serving People: Innovation through Co-creation, Liz Sanders, Northwestern University (talk) ............. 83
1.2.5.1.6 Exploring Co-design with Breastfeeding Mothers (paper) ............. 83
1.2.5.1.7 Exploring Co-design with Breastfeeding Mothers (talk) ............. 83
1.2.5.1.8 Heteromeration and its (dis)contents: The invisible division of labor between humans and machines (paper) ............. 83
1.2.5.1.9 RE lecture 5 Artifact Orientation (talk) ............. 83
1.2.5.1.10 RE lecture 6 Problem Vs Solution orientation (talk) ............. 83
1.2.5.1.11 RE lecture 7 Stakeholders (talk) ............. 83
1.2.5.1.12 Requirements Engineering - Business Case Analysis (talk) ............. 84
1.2.5.1.13 The Story of Co-Design by thinkpublic (talk) ............. 84
1.2.5.1.14 A Case Study of Toyota Unintended Acceleration and Software Safety (talk) ............. 84
1.2.5.1.15 The feasibility of quantifying the reliability of life-critical real-time software (paper) ............. 84
1.2.5.2 Eleventh Lecture -- 19 July 2018 ........................ 84
1.2.5.2.1 Chapter 6: Engineering and Operating Safer Systems Using STAMP: Applications to the real world: systems thinking applied to safety (book) ............. 85
1.2.5.2.2 Design Is (Empathic) – A Talk on Empathy, Humanity and Designing AI (talk) ............. 85
1.2.5.2.3 DPSIR = A Problem Structuring Method? An exploration from the “imagine” approach (paper) ............. 85
1.2.5.2.4 Methods & tools: the rich picture: a tool for reasoning about work context (paper) ............. 85
1.2.5.2.5 Practical hands-on accessibility testing (talk) ............. 85
1.2.5.2.6 RE lecture 9 System Vision (talk) ............. 85
1.2.5.2.7 Rich pictures: a means to explore the ‘sustainable mind’? (paper) ............. 85
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Chapter 1:</strong> Engineering a safer world: systems thinking applied to safety (book)</td>
<td>97</td>
</tr>
<tr>
<td>7</td>
<td><strong>Chapter 7:</strong> Who is the advocate?: stakeholders for sustainability: A Software System for Sustainability</td>
<td>99</td>
</tr>
<tr>
<td>8</td>
<td><strong>Chapter 8:</strong> Cheating more for less: Upward social comparisons motivate the poorly compensated to cheat (paper)</td>
<td>99</td>
</tr>
<tr>
<td>9</td>
<td><strong>Chapter 9:</strong> Ending Institutional Corruption</td>
<td>100</td>
</tr>
<tr>
<td>10</td>
<td><strong>Chapter 10:</strong> Fundamentals of Qualitative Research Methods: Focus Groups (Module 4) (talk)</td>
<td>101</td>
</tr>
<tr>
<td>11</td>
<td><strong>Chapter 11:</strong> Fundamentals of Qualitative Research Methods: Interviews (Module 3) (talk)</td>
<td>101</td>
</tr>
<tr>
<td>12</td>
<td><strong>Chapter 12:</strong> Fundamentals of Qualitative Research Methods: Scientific Rigor (Module 6) (talk)</td>
<td>101</td>
</tr>
<tr>
<td>13</td>
<td><strong>Chapter 13:</strong> Fundamentals of Qualitative Research Methods: What is Qualitative Research (Module 1) (talk)</td>
<td>101</td>
</tr>
<tr>
<td>14</td>
<td><strong>Chapter 14:</strong> Hacking iCorruption: Results from the Hackathon - May 2, 2015 (talk)</td>
<td>101</td>
</tr>
<tr>
<td>15</td>
<td><strong>Chapter 15:</strong> Learning is... Hard</td>
<td>101</td>
</tr>
<tr>
<td>16</td>
<td><strong>Chapter 16:</strong> Requirements Engineering - Elicitation Techniques (talk)</td>
<td>101</td>
</tr>
<tr>
<td>17</td>
<td><strong>Chapter 17:</strong> Robin Leach: Sustainable Software for a Digital Society (video)</td>
<td>103</td>
</tr>
<tr>
<td>18</td>
<td><strong>Chapter 18:</strong> What Do We Really Know About Meditation and Health?</td>
<td>103</td>
</tr>
<tr>
<td>19</td>
<td><strong>Chapter 19:</strong> The Dark Arts of Business Analysis: Requirements Gathering with Mind Maps for a Drupal Project (talk)</td>
<td>103</td>
</tr>
<tr>
<td>20</td>
<td><strong>Chapter 20:</strong> Second Class -- 27 May 2018</td>
<td>102</td>
</tr>
<tr>
<td>21</td>
<td><strong>Chapter 21:</strong> Accessibility for Everyone - An Evening with Laura Kalbag (video)</td>
<td>102</td>
</tr>
<tr>
<td>22</td>
<td><strong>Chapter 22:</strong> DevOpsDays Detroit 2017: Day 1 Keynote by Dr. Richard I. Cook (video)</td>
<td>103</td>
</tr>
<tr>
<td>23</td>
<td><strong>Chapter 23:</strong> On-Non-Functional Requirements (paper)</td>
<td>103</td>
</tr>
<tr>
<td>24</td>
<td><strong>Chapter 24:</strong> Patent Law: Sustainable Software for a Digital Society (video)</td>
<td>103</td>
</tr>
<tr>
<td>25</td>
<td><strong>Chapter 25:</strong> What Do We Really Know About Meditation and Health?</td>
<td>103</td>
</tr>
<tr>
<td>26</td>
<td><strong>Chapter 26:</strong> The Robotic Moment: Who Do We Become When We Talk to Machines? (talk)</td>
<td>104</td>
</tr>
<tr>
<td>27</td>
<td><strong>Chapter 27:</strong> The Dark Arts of Business Analysis: Requirements Gathering with Mind Maps for a Drupal Project (talk)</td>
<td>104</td>
</tr>
</tbody>
</table>
1.2.5.10.4 GOTO 2018 • A Strong Belief, Loosely Held: Bringing Empathy to IT • Nirmal Mehta (talk) .......................... 104
1.2.5.10.5 Open Source & Feelings 2015 - Designing with Empathy by Greg Tarnoff (talk) ......................................... 106
1.2.5.10.6 What is Permaculture? Bill Mollison, David Holmgren (talk) ................................................................. 106
1.2.5.11 Sixth Class -- 23 June 2018 ......................................................... 106
1.2.5.11.1 Chapter 17 Domestic Plant Guilds: A Software System for Sustainability: Software Engineering for Sustainability (thesis) ................................................................. 107
1.2.5.11.2 Contextual Enquiry (talk) ................................................................. 107
1.2.5.11.3 GOTO 2016 • Chaos & Intuition Engineering at Netflix • Casey Rosenthal (talk) ........................................... 107
1.2.5.11.4 Deprecating Simplicity • Casey Rosenthal (talk) .................................................................................. 107
1.2.5.11.5 Plant Guild Composer: An Interactive Online System to Support Back Yard Food Production (promo) .............. 107
1.2.5.12 Tenth Lecture -- 16 July 2018 .................................................... 108
1.2.5.12.1 Chapter 5: A Friendly Fire Accident – Engineering a safer world: systems thinking applied to safety (book) ....... 108
1.2.5.12.2 Chapter 18: An Assessment Technique for Sustainability: Applying the IMAGINE Approach to Software Systems: Software Engineering for Sustainability (thesis) .................................................. 108
1.2.5.12.3 Collaborative requirements negotiation with EasyWinWin (paper) .......................................................... 109
1.2.5.12.4 Creating Personas with Disabilities (paper) .................................................................................. 109
1.2.5.12.5 Developing a Tool for Testing Compatibility of Websites with ATs (paper) .............................................. 109
1.2.5.12.6 Evaluation of Accessibility Testing Methods. Which Methods Uncover What Type of Problems? (paper) ........... 109
1.2.5.12.7 Goal-directed concept acquisition in requirements elicitation (paper) ...................................................... 109
1.2.5.12.8 How to Read a Book by Mortimer Adler Animated Book Summary (talk) .................................................. 109
1.2.5.12.9 How To Read an Academic Paper (talk) ................................. 109
1.2.5.12.10 How to Read a Paper Efficiently (By Prof. Pete Carr) (talk) .................................................................. 110
1.2.5.12.11 How to Read a Research Paper (talk) ................................................................. 110
1.2.5.12.12 How to read a scientific paper (talk) .................................................................................. 110
1.2.5.12.13 How to Use Mind Mapping for Studying and Research (talk) .............................................................. 110
1.2.5.12.14 RE lecture 8 Goals (talk) ................................................................. 110
1.2.5.12.15 Requirements Engineering Goal Modeling (talk) .................................................................................. 110
1.2.5.12.16 Smart Cities and M3: Rapid Research, Meaningful Metrics and Co-Design (paper) ......................................... 110
1.2.5.12.17 SQ4R and Cornell Notes (talk) ................................................................. 110
1.2.5.12.18 Sustainability Indicators Past and Present: What Next? (paper) .......................................................... 111
1.2.5.12.19 Why Agent-Oriented Requirements Engineering (paper) ........................................................................ 111
1.2.5.13 Third Class -- 28 May 2018 ......................................................... 111
1.2.5.13.1 Characterizing the contribution of quality requirements to software sustainability (paper) ......................... 111
1.2.5.13.2 Evaluating the Accessibility and Usability of Blogging Platforms for Blind Users (paper) ......................... 111
1.2.5.13.3 George Loewenstein and the NEW New Economics of Information (talk) .................................................. 111
1.2.5.13.4 Requirements Engineering lecture 1: Overview (video) ................................................................. 111
1.2.5.14 Thirteenth Lecture -- 29 July 2018 ........................................... 112
1.2.5.14.1 "Google: Searching for a Circular Transition" by Kate Brandt (talk) ............................................................ 112
1.2.5.14.2 Allen Newell: Desires and Diversions : 1991 : Carnegie Mellon University (talk) ........................................ 112
1.2.5.14.3 Chapter 9: STPA: A New Hazard Analysis Technique -- Engineering a safer world: systems thinking applied to safety (book) ................................................................. 112
1.2.5.14.4 CHI 2011 SIGCHI Social Impact Award: Clayton Lewis & Alan Newell (talk) .................................................. 112
1.2.5.14.5 Design and the Digital Divide: Insights from 40 Years in Computer Support for Older and Disabled People (book) .................................................................................. 113
1.2.5.14.6 Global Public Inclusive Infrastructure (GPII) w/captions+description,no music (talk) ........................................ 113
1.2.5.14.7 In Pursuit of Mind: The Research of Allen Newell (paper) ........................................................................ 113
1.2.5.14.8 Kate Brandt; Google - WCS Talks 2016 (talk) .................................................................................. 113
1.2.5.14.9 On the analysis of human problem solving protocols (paper) ................................................................. 113
1.2.5.14.10 Protocol Analysis - Rev'd Edition: Verbal Reports as Data (book) ............................................................... 113
1.2.5.14.11 Reconciling software requirements and architectures with intermediate models (paper) .......................... 113
1.2.5.14.12 Refining Requirements - RE learning glass lecture 12 (talk) ................................................................. 113
1.2.5.14.13 Re-thinking Progress: The Circular Economy (talk) .................................................................................. 114
1.2.5.14.14 Scaling Requirements Engineering - learning glass lecture 11 (talk) .......................................................... 114
1.2.5.14.15 Soar : an architecture for general intelligence (paper) .............................................................................. 114
1.2.5.14.16 The prospects for psychological science in human-computer interactions (book) ....................................... 114
1.2.5.14.17 Unified theories of cognition (book) ........................................................................................................ 114
1.2.5.15 Twelfth Lecture -- 26 July 2018 .................................................. 114
1.2.5.15.1 S. Dennis Meadows - Perspectives on the Limits of Growth: It is too late for sustainable development (talk) ....... 114
1.2.5.15.2 A comparison of The Limits to Growth with 30 years of reality (paper) .......................................................... 115
1.2.5.15.3 Bypassing multi factor authentication - Shahmeer Amir, HackIT-2017 (talk) .................................................. 115
1.2.5.15.4 Chapter 7: Fundamentals -- Engineering a safer world: systems thinking applied to safety (book) ................. 116
1.2.5.15.5 Chapter 19: Supporting Physicians by REAS — Evaluating Requirements Engineering for Sustainability in the Medical Domain: Software Engineering for Sustainability (thesis) .................................................. 116
1.2.5.15.6 Guidelines for conducting and reporting case study research in software engineering (paper) ..................... 116
1.2.5.15.7 How The Economic Machine Works by Ray Dalio (talk) ................................................................. 116
1.2.5.15.8 On the (In)Security of Mobile Two-Factor Authentication (paper) ............................................................ 116
1.2.5.15.9 Overcoming Barriers to Medication Adherence for Chronic Diseases (talk) .................................................. 116
Overview of non-functional software requirements and requirements engineering

Understanding humans

1.3 Thinking about systems

1.3.1 A Definition of Systems Thinking: A Systems Approach (paper) .
1.3.2 An introduction to critical systems (talk) .
1.3.2.1 Critical National Infrastructure (talk) .
1.3.2.2 Critical systems engineering (talk) .
1.3.2.3 Emergent properties of sociotechnical systems (talk) .
1.3.2.4 Infrastructure dependability (talk) .
1.3.2.5 Infrastructure resilience (talk) .
1.3.2.6 Introducing sociotechnical systems (talk) .
1.3.2.7 Systems of systems (talk) .
1.3.2.8 System success and failure (talk) .
1.3.3 A Philosophical Look at System Dynamics (talk) .
1.3.3.1 Creating Models from Scratch (talk) .
1.3.3.2 System Dynamics Model: Kaibab Deer Population (talk) .
1.3.3.2.1 Deer population model in VENSIM (talk) .
1.3.3.2.2 Sustainability Modeling Spring 2013 Lecture 6: Modeling Population-Nature Interactions (talk) .
1.3.3.3 Systems: Overshoot and Collapse (talk) .
1.3.3.4 The Contribution of the Systems Sciences to the Humanities (paper) .
1.3.4 In A World of Systems (video) .
1.3.5 Introduction to System Dynamics: Overview (talk) .
1.3.5.1 Introduction to System Dynamics (MIT OpenCourseWare page) .
1.3.6 Norms that Govern: the Case of Wikipedia (talk) .
1.3.6.1 From Scaling to Practice: Real World Applications of Cities Research (talk) .
1.3.6.2 Software Engineering: Evolving Computer Programs (talk) .
1.3.6.3 The Complex Science of Cyberdefense: Computer Immunology (talk) .
1.3.6.4 The Myth of Common Sense: Why Everything that Seems Obvious Isn’t (talk) .
1.3.7 The Common Foundation Underlying Physical and Social Systems - Jay W. Forrester (talk) .
1.3.7.1 Applications of System Dynamics - Jay W. Forrester (talk) .
1.3.7.2 Jay Forrester: The Growth and Collapse of Markets (talk) .
1.3.8 Towards Acceptance of Socio-technical Systems – An Emphasis on the Requirements Phase (paper) .

1.4 Understanding humans

1.4.1 A Classification Model for Human Error in Collaborative Systems (thesis) .
1.4.1.1 A Systems Approach to Assessing, Interpreting and Applying Human Error Mishap Data to Mitigate Risk of Future Incidents in a Space Exploration Ground Processing Operations Environment (paper) .
1.4.1.2 The role of error in organizing behaviour (paper) .
1.4.2 ACT: a simple theory of complex cognition (paper) .
1.4.2.1 ACT-R home page (website) .
1.4.3.1 SOAR home page (web site) .
1.4.4 Electronic Checklist Support for Disaster Response (paper) .
1.4.5 Semantic services for information and management support in mass casualty incident scenarios (paper) .
1.4.6 The Use and Design of Flightcrew Checklists and Manuals (report) .
1.4.7 Self-Determination Theory (website) .

1.5 Overview of non-functional software requirements and requirements engineering

1.5.1 Are "Non-functional" Requirements really Non-functional? An Investigation of Non-functional Requirements in Practice (paper) .
1.5.2 Automated classification of non-functional requirements (paper) .
1.5.3 Design and evaluation of mass evacuation support systems using ontologies for improved situation awareness (Massey PhD Thesis) .
1.5.4 Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study (paper) .
1.5.5 Experimenting with Error Abstraction in Requirements Documents (paper) .
1.5.6 Extracting Non-Functional Requirements from Unstructured Text (U Waterloo thesis) .
1.5.7 Four dark corners of requirements engineering (paper) .
1.5.8 GaiaT: supporting the extraction of rights and obligations for regulatory compliance (paper) .
1.5.9 GDIA: Eliciting information requirements in emergency first response (paper) .
1.5.9.1 Developing Effective Standard Operating Procedures For Fire And EMS Departments (report) .
1.5.9.2 Emergency Scenario User Perspective in Public Safety Communications Systems (paper) .
1.5.9.3 Evaluating the Reliability of Emergency Response Systems for Large-Scale Incident Operations (report) .
1.5.9.4 Information systems architecture for fire emergency response (paper) .
1.5.10 Getting to the Shallows: Facilitating Sensemaking in Requirements Engineering (paper) .
1.5.11 Goal-Based Requirements Analysis (paper) .
1.5.11.1 Goal Identification and Refinement in the Specification of Information Systems (Georgia Tech thesis) .

1.6 Music for a Quiet Day .
1.7 Notes past the eleventh class .
1.8 Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018) .
1.9 Searching the Computer Science Literature .
| 1.5.11.2 Representing and Using Nonfunctional Requirements: A Process-Oriented Approach (paper) | 133 |
| 1.5.12 Investigation of user requirements in the emergency response sector: the Dutch case (paper) | 133 |
| 1.5.12.1 Disaster Robotics (book) | 133 |
| 1.5.13 Missing Requirements Information and its Impact on Software Architectures: A Case Study (UWO Thesis) | 134 |
| 1.5.13.1 Maps of Lessons Learnt in Requirements Engineering (UWO Thesis) | 134 |
| 1.5.13.2 Regulatory Compliance-oriented Impediments and Associated Effort Estimation Metrics in Requirements Engineering for Contractual Systems Engineering Projects (UWO Thesis) | 134 |
| 1.5.14 Quality Requirements in Agile as a Knowledge Management Problem: More than Just-in-Time (paper) | 134 |
| 1.5.14.1 Requirements and testing: seven missing-link myths (paper) | 134 |
| 1.5.16 Requirements engineering: a good practice guide (book) | 134 |
| 1.5.16.1 An introduction to Requirements Engineering (talk) | 134 |
| 1.5.17 Requirements engineering: a roadmap (paper) | 135 |
| 1.5.18 Requirements Engineering (journal) | 135 |
| 1.5.20 Requirements in the wild: How small companies do it (talk) | 135 |
| 1.5.21.1 The Prevalence and Impact of Persistent Ambiguity in Software Requirements Specification Documents (U Waterloo thesis) | 135 |
| 1.5.21.1 The Impact of Domain Knowledge on the Effectiveness of Requirements Engineering Activities (U Waterloo Thesis) | 135 |
| 1.5.22 Usability of requirements techniques: a systematic literature review (paper) | 136 |
| 1.5.23 User requirements for a collective intelligence emergency response system (paper) | 136 |
| 1.5.23.1 Semantic Representation of Process and Service Compliance — A Case Study in Emergency Planning (paper) | 136 |
| 1.5.24 Voting Systems and the Condorcet Paradox | Infinite Series (talk) | 136 |
| 1.6 From requirements to design | 136 |
| 1.6.1 From Non-Functional Requirements to Design through Patterns (paper) | 136 |
| 1.7 How safety requirements figure into software systems | 136 |
| 1.7.1 Air Force System Safety Handbook (report) | 138 |
| 1.7.2 Analysing the Safety of a Software Development Process (paper) | 138 |
| 1.7.3 An Approach to Designing Safe Embedded Software (paper) | 138 |
| 1.7.4 Design and Operating Safety Systems: The Missing Link - SOA and SM (talk) | 138 |
| 1.7.3.2 Engineering a safer world: systems thinking applied to safety (book) | 138 |
| 1.7.3.2.1 A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on STPA. (paper) | 139 |
| 1.7.3.3 Introduction to: Systems Theoretic Accident Model & Processes (STAMP) WEBINAR REPLAY (talk) | 139 |
| 1.7.3.4 Software Challenges In Achieving Space Safety (paper) | 139 |
| 1.7.3.5 The Need for a Paradigm Shift in Safety and Cyber Security (talk) | 139 |
| 1.7.3.6 Integrated approach to safety and security based on Systems theory (paper) | 139 |
| 1.7.4.1 The SEMA referential framework: Avoiding ambiguities in the terms “security” and “safety” (paper) | 140 |
| 1.7.5 Building a safety architecture pattern system (paper) | 140 |
| 1.7.6 Coping with Defective Software in Medical Devices (paper) | 140 |
| 1.7.7 Design, Implementation, and Verification of a Reactor Protection System Using HFC6000 (UWO Thesis) | 140 |
| 1.7.7.1 Predictive Shutdown Systems for Nuclear Power Plants (UWO Thesis) | 140 |
| 1.7.8 Drivers’ needs and safety systems (paper) | 140 |
| 1.7.9 Early Concept Development and Safety Analysis of Future Transportation Systems (paper) | 140 |
| 1.7.9.1 Engineering financial safety-net systems: a system-theoretic case study from the financial crisis (paper) | 140 |
| 1.7.11 Engineering safety-related requirements for software-intensive systems (paper) | 141 |
| 1.7.12 Fail-safe testing of safety-critical systems: a case study and efficiency analysis (paper) | 141 |
| 1.7.13 Inside risks: a risks-related bookshelf (paper) | 141 |
| 1.7.14 Integration between requirements engineering and safety analysis: A systematic literature review (paper) | 141 |
| 1.7.14.1 Requirements engineering for safety-critical systems: A systematic literature review (paper) | 141 |
| 1.7.15 International Conference on Computer Safety, Reliability, and Security (proceedings) | 141 |
| 1.7.16 Is safety a subject for science? (paper) | 142 |
| 1.7.17 Opening the sources of accountability (paper) | 142 |
| 1.7.18 Safe operation as a social construct (paper) | 142 |
| 1.7.19 Safety-Critical Systems - Professor Martyn Thomas CBE (talk) | 142 |
| 1.7.20 Safety Issues in Advanced AI (talk) | 142 |
| 1.7.20.1 Facing the unknown: the future of humanity (talk) | 142 |
| 1.7.21 Success and failure: human as hero — human as hazard (paper) | 142 |
| 1.7.22 Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems (paper) | 143 |
| 1.7.22.1 Assessing Dependability with Software Fault Injection: A Survey (paper) | 143 |
| 1.7.22.2 Why do some (weird) people inject faults? (paper) | 143 |
| 1.7.23 The ethics of safety-critical systems (paper) | 143 |
| 1.7.23.1 ACM Code of Ethics | 143 |
| 1.7.24 The Safety of Software – Constructing and Assuring Arguments (U of York thesis) | 143 |
| 1.7.25 The unruly power grid (paper) | 143 |
| 1.7.25.1 Notes on Ontario and 2003 North American Power Failure | 144 |
| 1.7.26 Towards state-of-the-art and future trends in testing of active safety systems (paper) | 144 |
| 1.7.27 Verification of safety-critical software (paper) | 144 |
| 1.8 How accessibility requirements figure into software systems | 144 |
| 1.8.1 8 80 Cities (Toronto web site) | 146 |
| 1.8.2 Accessibility, usability and universal design—positioning and definition of concepts describing person-environment relationships (paper) | 147 |
| 1.8.2.1 Bridging the gap: between accessibility and usability (paper) | 147 |
| 1.8.2.2 The relationship between accessibility and usability of websites (paper) | 147 |
1.8.3 Accessibility for Sakai (open source educational software) ........................................ 147
1.8.4 Accessibility London Ontario Canada (web site) ............................................................. 147
1.8.5 Accessibility of Atlassian Products .................................................................................. 147
1.8.6 Accessibility Western (web site) ....................................................................................... 148
1.8.7 ACE: A Colour Palette Design Tool for Balancing Aesthetics and Accessibility (paper) .... 148
1.8.8 ACM Transactions on Accessible Computing (TACCESS) (journal web site) ......................... 148
1.8.9 A Knowledge-Based Framework for Improving Accessibility of Web Sites (paper) .......... 148
1.8.10 An Introduction to the Theory and Measurement of Self Determination (video) ............. 149
1.8.11 ASSETS 2012 SIGACCESS Conference on Computers and Accessibility (Proceedings) ....... 149
1.8.12 A Support System for Graphics for Visually Impaired People (UWO Thesis) .................... 149
1.8.12.1 Accessibility of E-Commerce Websites for Vision Impaired Persons (UWO Thesis) .......... 149
1.8.12.2 Examination of the Usability of the IPAD Among Older Adult Consumers (UWO Thesis) .. 149
1.8.12.3 Redesign of Johar: a framework for developing accessible applications (UWO Thesis) ........ 149
1.8.13 Best Practices for Teaching Accessibility in University Classrooms: Cultivating Awareness, Understanding, and Appreciation for Diverse Users (paper) ...................................................... 150
1.8.14 Beyond Web Content Accessibility Guidelines: Expert Accessibility Reviews (paper) ........ 150
1.8.15 Design and Study of Emotions in Virtual Humans for Assistive Technologies (U Waterloo thesis) 150
1.8.15.1 An Assistive Handwashing System with Emotional Intelligence (U Waterloo thesis) ........ 150
1.8.16 Ensuring Universal Design in ICT-Solutions - Towards Identifying Critical Success Factors (Thesis) 150
1.8.17 Entangled With Books: Two Moments In the Evolution of Accessibility and the Notion of Universal Design (thesis) ................................................................. 150
1.8.18 Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities (U Pittsburgh Thesis) ................................................. 150
1.8.19 Exploring language technologies to provide support to WCAG 2.0 and E2R guidelines (paper) 151
1.8.20 Finding Novice Web Writing in Making Images: Designing Using Templates (paper) ............ 151
1.8.21 Inclusive Design (talk) ....................................................................................................... 151
1.8.21.1 Design Meets Disability (talk) (book) ............................................................................. 151
1.8.21.2 Haben Girma: "Designing Technology with Accessibility in Mind" | Talks at Google (talk) ... 151
1.8.21.3 Inclusive design of ICT: The challenge of diversity (U Oslo thesis) ................................. 151
1.8.21.4 Making a difference. One app at a time: Disability, Distinction, and Domestic Access (talk) 152
1.8.21.5 Mobile Gesture-Based User Interfaces for People with Disabilities (talk) ......................... 152
1.8.21.6 User characteristics and the effectiveness of inclusive design for older users: guidelines for designers (Humboldt-Universität zu Berlin thesis) ......................................................... 152
1.8.21.7 User-Sensitive Inclusive Design (paper) ......................................................................... 152
1.8.22 Interviews and Observation of Blind Software Developers at Work to Understand Code Navigation Challenges (paper) .................................................................................. 152
1.8.23 Iterative Design and Field Trial of an Aphasia-Friendly Email Tool (paper) ......................... 152
1.8.24 List of Accessibility Blogs (link) ....................................................................................... 152
1.8.24.1 A11Y Project (resources) .................................................................................................. 152
1.8.24.2 Gov UK Accessibility blog (blog) .................................................................................... 153
1.8.24.3 The Accessibility Hub (blog, Canada) ............................................................................. 153
1.8.24.4 Web Axe -- blog and podcast on web accessibility (blog) ............................................. 153
1.8.25 Making the web easier to see with opportunistic accessibility improvement (paper) ............. 153
1.8.26 Measuring website usability for visually impaired people—a modified GOMS analysis (paper) 153
1.8.27 Mental Models of Chinese and German Users and Their Implications for MMI: Experiences from the Case Study Navigation System (thesis) ................................................................. 153
1.8.28 Audiences, memories, expectations and confidence experienced by older users (paper) ........ 153
1.8.29 Principles and Practice of Assistive Technology (MIT OpenCourseware) ............................ 154
1.8.30 Rupal Patel: Synthetic Voices as unique as fingerprints (talk) ........................................... 154
1.8.30.1 "Speech Synthesis," Kim Silverman (talk) ........................................................................ 154
1.8.30.2 Prof. Simon King - Using Speech Synthesis to give Everyone their own Voice (talk) ......... 154
1.8.30.3 Roger Ebert: Remaking My Voice .................................................................................. 154
1.8.31 Section 508 Amendment to the Rehabilitation Act of 1973 (US Accessibility Legal Requirement) (links) 154
1.8.32 Systems Thinking in Design: Service Design and self-Services (paper) .............................. 154
1.8.33 Teach or Design? How Older Adults’ Use of Ticket Vending Machines Could Be More Effective (paper) ................................................................. 155
1.8.33.1 Sustainable E2 mobility services for elderly people — Platform system architecture (paper) 155
1.8.33.2 Usability, accessibility and ambient-assisted living: a systematic literature review (paper) ... 155
1.8.34 The challenges of Web accessibility: The technical and social aspects of a truly universal Web (paper) ................................................................................................. 155
1.8.34.1 Auditing Web accessibility: The role of interest organizations in promoting compliance through certification (paper) .......................................................................................... 155
1.8.34.2 Blurred lines: Accessibility, disability, and conceptual limitations (paper) ....................... 155
1.8.34.3 Code in action: Closing the black box of WCAG 2.0. A Latourian reading of accessibility (paper) 155
1.8.34.4 Disability, human rights, and social justice: The ongoing struggle for online accessibility and equality (paper) ................................................................. 155
1.8.35 Exploring the accessibility of banking and finance systems for blind users (paper) ............... 156
1.8.36 Stanford Seminar - Building an Accessible Web (talk) ...................................................... 156
1.8.36.1 The Effectiveness of the Web Accessibility Audit as a Motivational and Educational Tool in Inclusive Web Design (U Dundee thesis) ......................................................... 156
1.8.36.2 Creating an accessibility guideline for the 2020s (paper) ............................................... 156
1.8.36.3 The Effect of Font Type on Screen Readability by People with Dyslexia (paper) ................. 156
1.8.36.4 Universal Access in the Information Society: International Journal (journal) ..................... 156
1.8.37 Universal Design (talk) ...................................................................................................... 156
1.8.38 Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users (paper) ................................................................. 156
1.8.39 Web Content Accessibility Guidelines (WCAG) 2.0 (web document) ................................. 157
1.8.40 Why the Dvorak keyboard didn't take over the world (talk) .............................................. 157
1.8.41 YOW! West 2017 Ann Mwangi - Accessibility: Is it another checkbox to be ticked? (talk) .................................................. 157
1.8.41.1 "Making the Digital and Physical Worlds Accessible for People with Disabilities," Eve Andersson (talk) ...... 157
1.8.41.2 Google I/O 2011: Creating Accessible Interactive Web Apps using HTML5 (talk) ...................................................... 157
1.8.41.3 Google I/O 2015 - Improve your Android app's accessibility (talk) ................................................................. 157
1.8.41.3.1 Braille Displays (note) ........................................................................................................................................ 157
1.8.41.3.2 Russell Couper : CANUTE, The worlds first working refreshable multilinie braille display. (talk) ............... 158
1.8.41.3.3 Switch Access for Android (talk) ......................................................................................................................... 158
1.8.41.8.1 2015: Automated Accessibility Testing for Android Applications (talk) ......................................................... 158
1.8.41.8.5 Inclusive design and testing: Making your app accessible - Google I/O 2016 (talk) ........................................... 158
1.8.41.6 Marcy Sutton: 30 Minutes or Less: The Magic of Automated Accessibility Testing | JSConf US 2015 (talk) ...... 158
1.8.41.7 Paul J. Adam: Web Accessibility Testing Tools and Techniques Paul Bohman (talk) ............................................. 158
1.8.41.8 Telekinetic Control of Android and Other Wonders of the Modern World (talk) ................................................... 158
1.8.41.8.1 Communicating Through a Brain-Computer Interface: Towards the Assessment of Quality of Life in Minimally Conscious and Covertly Aware Vegetative State Patients (UWO Thesis) .................................................. 158
1.8.41.8.2 SpeakYourMind: Aaron typing with his eyes! Powered by EyeTribe and Microsoft (demo) .......................... 158
1.8.41.8.3 Tecla Access - Reading an e-book (talk) .................................................................................................................. 159
1.8.41.8.4 Towards the Development of a Wearable Tremor Suppression Glove (UWO Thesis) ........................................ 159
1.8.41.9 Testing for Accessibility Linux.conf.au 2014 -- Perth, Australia (talk) ............................................................ 159

1.9 How sustainability requirements figure into software systems .................................................................................. 159
1.9.1 Academic ICT and Green Cyberinfrastructure (talk) ................................................................................................. 161
1.9.2 Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper) ..................................... 161
1.9.2.1 Toward Architecture Knowledge Sustainability: Extending System Longevity (paper) ......................................... 161
1.9.2.1.1 Identifying Architectural Bad Smells (paper) ........................................................................................................... 161
1.9.2.1.2 Relating Architectural Decay and Sustainability of Software Systems (paper) .................................................. 161
1.9.3 A systematic literature review for software sustainability measures (paper) .......................................................... 162
1.9.4 Cyberinfrastructure Software Sustainability & Reusability Workshop - Software and the Long Haul (talk) ....... 162
1.9.4.1 Creating Sustainable Cyberinfrastructures (talk) ......................................................................................................... 162
1.9.4.2 Cultivating Sustainable Software for Research (talk) .............................................................................................. 162
1.9.4.3 Cyberinfrastructure Software Sustainability & Reusability Workshop - An Industry View (talk) ....................... 162
1.9.4.4 Cyberinfrastructure Software Sustainability & Reusability Workshop - Software Reuse (talk) ............................ 162
1.9.4.5 Cyberinfrastructure Software Sustainability & Reusability Workshop - Sustainable Software (talk) .................. 162
1.9.4.6 Cyberinfrastructure Software Sustainability & Reusability Workshop - Welcome Address (talk) ......................... 162
1.9.4.7 Cyberinfrastructure Software Sustainability and Reusability: Report from an NSF-funded workshop (report) ... 163
1.9.5 Framing sustainability as a property of software quality (paper) ............................................................................. 163
1.9.6 GreenOracle: estimating software energy consumption with energy measurement corpora (paper) .................. 163
1.9.7 Improving Software Sustainability through Data-driven Technical Debt Management (talk) .......................... 163
1.9.8 Information systems for the age of consequences (paper) ........................................................................................ 163
1.9.8.1 Abstraction, indirection, and Sevareid's Law: Towards benign computing (paper) ............................................. 163
1.9.8.2 Collapse (& other futures) software engineering (paper) ....................................................................................... 163
1.9.8.3 Computing within limits and ICTD (paper) ............................................................................................................... 164
1.9.8.4 Limits and sustainable interaction design: Obsolescence in a future of collapse and resource scarcity (paper) ... 164
1.9.8.5 Preliminary thoughts on a taxonomy of value for sustainable computing (paper) ................................................. 164
1.9.9 International Workshop on Green and Sustainable Software (Proceedings web sites) ...................................... 165
1.9.10 Jumpstart sustainability in seminars: hands-on experiences in class (paper) .......................................................... 165
1.9.10.1 A Curriculum on Sustainable Information Communication Technology (paper) ............................................ 165
1.9.10.2 An international Master's program in green ICT as a contribution to sustainable development (paper) ........... 165
1.9.10.3 Educational aspects of sustainable development analysis: computational models and software (paper) ........ 165
1.9.10.4 Infusing sustainability into software engineering education: Lessons learned from capstone projects (paper) ... 165
1.9.11 Lynn Bartram - Human-Centred Systems for Sustainable Living (talk) .............................................................. 165
1.9.11.1 Patricia Lago: Sustainable Software for a Digital Society (talk) ................................................................. 165
1.9.11.2 Performativity in sustainable interaction: the case of seasonal grocery shopping in ecofriends (paper) .......... 166
1.9.13 Preparing research projects for sustainable software engineering in society (paper) ........................................... 166
1.9.14 Quality Metrics for Sustainability: The Operational Energy Use of Application Software (Utrecht U thesis) .... 166
1.9.15 Requirements: Sustainability and Requirements: A Manifesto ieeeComputerSociety (talk) ............................... 166
1.9.15.1 Sustainability Design and Software: The Karlskrona Manifesto (paper) .............................................................. 166
1.9.16 Security, Nonfunctional Requirements for the 21st Century (paper) ...................................................................... 166
1.9.16.1 Characterising sustainability requirements: a new species, red herring, or just an odd fish? (paper) ............... 167
1.9.17 Software Engineering Aspects of Green and Sustainable Software: A Systematic Mapping Study (paper) .... 167
1.9.18 Software Engineering for Sustainability (thesis) ...................................................................................................... 168
1.9.19 So Much Money for So Little Capability: The Reality of Sustaining DoD Software Systems (slides for talk) ....... 167
1.9.19.1 Measuring software sustainability (paper) .................................................................................................................. 167
1.9.19.2 Sustainability guidelines for long-living software systems (paper) ................................................................. 167
1.9.20 Sustainability debt: a portfolio-based approach for evaluating sustainability requirements in architectures (paper) 167
1.9.21 Sustainability design in requirements engineering: state of practice (paper) ......................................................... 168
1.9.22 Sustainable software design (paper) ......................................................................................................................... 168
1.9.23 Systematic mapping study on software engineering for sustainability (SE4S) (paper) ........................................... 168
1.9.24 The Club of Rome and Limits To Growth: Achieving the Best Possible Future (talk) ............................................. 168
1.9.24.1 Aggregating Evidence in Climate Science: Consilience, Robustness and the Wisdom of Multiple Models (UWO Thesis) (paper) 168
1.9.24.2 A System Dynamics Based Integrated Assessment Modelling of Global-Regional Climate Change: A Model for Analyzing the Behaviour of the Social-Energy-Economy-Climate System (UWO Thesis) 168
1.9.24.3 A Systems Approach to Modelling the Effects of Climate Change on Agroforestry: A Case Study in Western Tanzania (UWO Thesis) 169
1.9.24.4 Biopolitics of Climate Change: Carbon Commodities, Environmental Profanations, and the Lost Innocence of Use-Value (UWO Thesis) 169
1.9.24.5 On the Economics of Climate Change and its Effects (UWO Thesis) 169
1.9.24.6 Thomas HAGER - Haber-Bosch: The Discovery that changed the World (talk) 169
1.9.24.6.1 Keynote Address: Mr. Thomas Hager (talk) 171
1.9.25 The Management and Sustainability of Organizational Change in Primary Care Adoption of Electronic Medical Record Systems (McMaster thesis) 169
1.9.26 Three Essays on Sustainable Operations Management (UWO Thesis) 169
1.9.27 Towards a definition of sustainability in and for software engineering (paper) 169
1.9.28 Who is the advocate?: stakeholders for sustainability (paper) 169
1.10 How other non-functional requirements figure into software systems 170
1.10.1 Acceptability as a non-functional software requirement 170
1.10.1.1 On design-oriented research and digital learning materials in higher education (Wageningen PhD Thesis) 170
1.10.1.2 Personalized adaptation in pervasive systems via non-functional requirements (paper) 170
1.10.1.3 Requirements-Aware Systems: A Research Agenda for RE for Self-Adaptive Systems (paper) 170
1.10.1.4 The influence of student characteristics on the use of adaptive e-learning material (paper) 171
1.10.1.4.1 Bioprocess-Engineering Education with Web Technology (U of Wageningen PhD Thesis) 171
1.10.1.4.2 Designing an adaptive web-based learning system based on students' cognitive styles identified online (paper) 171
1.10.1.4.3 Proteus: A Tutor-Friendly Adaptive Tutoring System (paper) 171
1.10.1.5 Towards Effective Research-Paper Recommender Systems and User Modeling based on Mind Maps (PhD Thesis) 171
1.10.1.6 User modelling for adaptive computer systems: a survey of recent developments (paper) 171
1.10.2 Learnability as a non-functional requirement 171
1.10.2.1 A survey of software learnability: metrics, methodologies and guidelines (paper) 171
1.10.3 Privacy as a non-functional software system requirement 172
1.10.3.1 Office of the Privacy Commissioner of Canada (web site) 172
1.10.3.1.1 Information and Privacy Commissioner of Ontario (web site) 172
1.10.3.1.2 Time to modernize tools to protect personal information, Privacy Commissioner urges in annual report (news release) 172
1.10.3.2 Privacy in Cooperative Distributed Systems: Modeling and Protection Framework (UWO Thesis) 172
1.10.3.2.1 Protecting Personal Private Information in Collaborative Environments (UWO Thesis) 172
1.10.3.3 The Ad Contrarian Blog 173
1.10.3.4 Availability as a non-functional system requirement 173
1.10.4.1 Erlang (paper) 173
1.10.4.1.1 Making reliable distributed systems in the presence of software errors (thesis) 173
1.10.4.2 Evolution of fault tolerance (talk) 173
1.10.4.2.1 Designing Fault Tolerant Applications (Amazon webinar) 173
1.10.4.2.2 LambdaConf 2015 - Fault Tolerance on the Cheap Making Systems That Probably Won't Fall Over (talk) 173
1.10.4.2.3 Ruby Conf 2013 - Fault Tolerant Data: Surviving the Zombie Apocalypse (talk) 173
1.10.4.3 Reliability Models and Algorithms Applied to Smartphone Application (UWO Thesis) 174
1.10.4.4 Software Reliability Engineering (ISSRE), International Symposium on (proceedings) 174
1.10.5 Security as a non-functional software system requirement 174
1.10.5.1 "Building Secure Cultures" by Leigh Honeywell (talk) 175
1.10.5.1.1 "Incident insights from NASA, NTSB, and the CDC" by Emil Stolarsky (talk) 175
1.10.5.2 ACM Transactions on Privacy and Security (TOPS) (Journal web site) 176
1.10.5.3 Analyzing Regulatory Rules for Privacy and Security Requirements (paper) 176
1.10.5.3.1 A survey of compliance issues in cloud computing (paper) 176
1.10.5.3.2 Identifying vulnerabilities and critical requirements using criminal court proceedings (paper) 176
1.10.5.3.3 Legally "reasonable" security requirements: A 10-year FTC retrospective (paper) 176
1.10.5.4 An empirical study on the effectiveness of security code review (paper) 176
1.10.5.5 Applying systems thinking to healthcare data cybersecurity (MIT thesis) 176
1.10.5.6 A systems perspective on cybersecurity in the cloud : frameworks, metrics and migration strategy (MIT thesis) 176
1.10.5.7 Building secure software: how to avoid security problems the right way (book) 177
1.10.5.8 Computer security in the real world (paper) 177
1.10.5.9 Cryptanalytic Attacks on Pseudorandom Number Generators (paper) 177
1.10.5.10 Cryptographic voting protocols: a systems perspective (paper) 177
1.10.5.11 Cyber Threats and Cyber Security (talk) 177
1.10.5.12 Designing security into software (MIT thesis) 177
1.10.5.13 Emergent Properties & Security: The Complexity of Security as a Science (paper) 177
1.10.5.14 Far from Home: A Travel Security Guide for Government Officials (Canada) (report) 177
1.10.5.15 Formal methods for web security (paper -- survey) 178
1.10.5.16 How Do System Administrators Resolve Access-Denied Issues in the Real World? (paper) 178
1.10.5.17 Information Security at Western (Web Site) 178
1.10.5.18 Maroochy water breach (talk) 178
1.10.5.19 Measuring the security impacts of password policies using cognitive behavioral agent-based modeling (paper) 178
1.10.5.19.1 General Requirements of a Hybrid-Modeling Framework for Cyber Security (paper) 178
1.10.5.20 Model Checking An Entire Linux Distribution for Security Violations (paper) 178
1.10.5.21 New Cyber Frontiers (video) ................................................................. 179
1.10.5.22 One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation (paper) ........................................... 179
1.10.5.23 One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques (paper) ......................................... 179
1.10.5.24 Privacy Impacts of IoT Devices: A SmartTV Case Study (paper) ....................................................................................... 179
1.10.5.25 Risk Assessment for IoT: a system evaluation of the smart home and its cybersecurity imperative (MIT thesis) .............................. 179
1.10.5.26 Schneier on Security (blog) ................................................................ 179
1.10.5.27 Security collapse in the HTTPS market (paper) .................................. 179
1.10.5.27.1 How Internet Resources Might Be Helping You Develop Faster but Luring You Closer (paper) .................................................. 180
1.10.5.27.2 The most dangerous code in the world: validating SSL certificates in non-browser software (paper) ................................. 180
1.10.5.27.3 The web won’t be safe or secure until we break it (paper) ................ 180
1.10.5.28 Security engineering: a guide to building dependable distributed systems (book) ............................................................... 180
1.10.5.29 Selecting Security Patterns that Fulfill Security Requirements (paper) ...................................................................................... 180
1.10.5.30 Software and the mundane management of air travel (paper) ............... 180
1.10.5.31 Software-defined Networking enabled Resource Management and Security Provisioning in 5G Heterogeneous Networks (UWO Thesis) ........................................................................ 181
1.10.5.31.1 An Access Control Model for NoSQL Databases (UWO Thesis) ...... 181
1.10.5.31.2 Secure OFDM System Design for Wireless Communications Hao Li, The University of Western Ontario (UWO Thesis) ........................ 181
1.10.5.32 Software security testing (paper) ........................................................ 181
1.10.5.32.1 An overview of vulnerability assessment and penetration testing techniques (paper) ............................................................. 181
1.10.5.32.2 Comprehensive study of web application attacks and classification (paper) .............................................................................. 181
1.10.5.32.3 Mitigating program security vulnerabilities: web approaches and challenges (paper) .............................................................. 181
1.10.5.32.4 Open-Source Security Testing Methodology Manual (document) ...................................................................................... 182
1.10.5.32.5 Security Testing Fundamentals - Ken van Wyk (talk) ..................... 182
1.10.5.32.6 Software penetration testing (paper) ................................................ 182
1.10.5.32.7 Software Vulnerability Discovery Techniques: A Survey (paper) ..................................................................................... 182
1.10.5.32.8 The Red Book: A Roadmap for Systems Security Research (document) .............................................................................. 182
1.10.5.32.9 Why security testing is hard (paper) .................................................. 182
1.10.5.33 Survey on JavaScript security policies and their enforcement mechanisms (paper) ............................................................... 182
1.10.5.34 Surviving the Web: A Journey into Web Session Security (paper) ........... 183
1.10.5.35 The Internet of Things: Implications for Consumer Privacy under Canadian Law (paper) ......................................................... 183
1.10.5.36 Timing Analysis of Keystrokes and Timing Attacks on SSH (paper) ...... 183
1.10.5.36.1 Practical Keystroke Timing Attacks in Sandboxed JavaScript. (paper) ............................................................... 183
1.10.5.37 Using a systems-theoretic approach to analyze cyber attacks on cyber-physical systems (MIT thesis) ........................................ 183
1.10.5.38 Using STAMP to understand recent increases in malicious software activity (MIT thesis) ...................................................... 183
1.10.5.39 Verifying computations without reexecuting them (paper) ................... 183
1.10.5.40 What Every Engineer Needs to Know About Security and Where to Learn It (talk) ............................................................ 184
1.10.5.41 Why cryptosystems fail (paper) ............................................................ 184
1.10.5.41.1 EMV: why payment systems fail (paper) ......................................... 184
1.10.5.42 Why information security is hard - an economic perspective (paper) ........ 184
1.10.6 Technology Acceptance as a non-functional requirement ......................... 184
1.10.6.1 A computational literature review of the technology acceptance model (paper) ........................................................................... 184
1.10.6.2 Technology Acceptance and User Experience: A Review of the Experiential Component in HCI (paper) .................................. 184
1.10.6.3 Technology acceptance model: a literature review from 1986 to 2013 (paper) ......................................................................... 184
1.10.6.4 The Technology Integration Model (TIM): Predicting the continued use of technology (paper) .................................................. 185
1.10.7 Usability as a non-functional requirement .................................................. 185
1.10.7.1 Architecting for usability: a survey (paper) ............................................. 185
1.10.7.2 A Strategic Approach to Metrics for User Experience Designers (paper) ...................................................................................... 186
1.10.7.3 IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use (paper) ......................... 186
1.10.7.4 Journal of Usability Studies (journal) ..................................................... 186
1.10.7.5 Measuring the user experience on a large scale: user-centered metrics for web applications (paper) .............................................. 186
1.10.7.6 Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience (paper) ......................... 186
1.10.7.7 Predictive human performance modeling made easy (paper) .................. 186
1.10.7.7.1 Deploying CogTool: integrating quantitative usability assessment into real-world software development (paper) ....................... 186
1.10.7.8 Reverse engineering of content to find usability problems: a healthcare case study (paper) ....................................................... 186
1.10.7.9 SUS: a retrospective (paper) ................................................................. 187
1.10.7.10 The Usability Metric for User Experience (paper) ................................. 187
1.10.7.11 Why Johnny can’t encrypt: A usability evaluation of PGP 5.0 (paper) ...................................................................................... 187
1.10.7.11.1 Towards Enhanced Usability of IT Security Mechanisms - How to Design Usable IT Security Mechanisms Using the Example of Email Encryption (paper) ........................................ 187
1.11 Writing papers in computer science ................................................................. 187
1.11.1 Case studies synthesis: a thematic, cross-case, and narrative synthesis worked example (paper) ......................................................... 188
1.11.2 Guidelines for snowballing in systematic literature studies and a replication in software engineering (paper) ................................. 188
1.11.2.1 Experiences from using snowballing and database searches in systematic literature studies (paper) ........................................... 189
1.11.2.2 Second-generation systematic literature studies using snowballing (paper) .............................................................................. 189
1.11.2.3 Systematic literature studies: database searches vs. backward snowballing (paper) ................................................................. 189
1.11.2.4 Using Forward Snowballing to update Systematic Reviews in Software Engineering (paper) ...................................................... 189
1.11.3 Performing systematic literature reviews in software engineering (paper) ...................................................................................... 189
1.11.3.1 A critical analysis of studies that address the use of text mining for citation screening in systematic reviews (paper) ......................... 189
1.11.3.2 A critical appraisal of systematic reviews in software engineering from the perspective of the research questions asked in the reviews (paper) .................................................. 189
1.11.3.3 Experience-based guidelines for effective and efficient data extraction in systematic reviews in software engineering (paper) ................................................................. 190
1.11.3.4 Outcomes of a community workshop to identify and rank barriers to the systematic literature review process (paper) ................................................................................. 190
1.11.3.5 Quality assessment of systematic reviews in software engineering: a tertiary study (paper) ........... 190
1.11.3.6 Strength of evidence in systematic reviews in software engineering (paper) ........................................ 190
1.11.3.7 Systematic literature reviews and undergraduate research (paper) .............................................. 190
1.11.3.8 Systematic review in software engineering: where we are and where we should be going (paper) .... 190
1.11.4 Presenting software engineering results using structured abstracts: a randomised experiment (paper) ...... 190
1.11.5 Software engineering article types: An analysis of the literature (paper) ........................................... 191
1.11.6 Systematic mapping studies in software engineering (paper) .......................................................... 191
1.11.6.1 Guidelines for conducting systematic mapping studies in software engineering: An update (paper) ...... 191
1.11.6.2 Improving Systematic Mapping Reviews (paper) ............................................................................ 191
1.11.6.3 Using mapping studies as the basis for further research - A participant-observer case study (paper) ...... 191
1.11.7 The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature (paper) ................................................................ 191
1.11.8 Using qualitative metasummary to synthesize empirical findings in literature reviews (paper) ........... 191
1.11.9 Writing for synthesis of evidence in empirical software engineering (paper) .................................... 192
1.11.10 Framing a research study (paper) ..................................................................................................... 192
1.11.10.1 Research methods in computing: what are they, and how should we teach them? (paper) .............. 192
1.11.11 YOW! Nights October 2014 Damian Conway - Instantly Better Presentations (talk) ......................... 192
1.11.11.1 Evaluating oral presentations (paper) ............................................................................................. 192
1.11.11.2 What Does It Mean to “Measure Performance”? (paper) ............................................................... 192
1.11.12 Writing for Computer Science [electronic resource] / by Justin Zobel (book) ...................................... 192
1.11.12.1 How to Write a Better Thesis [electronic resource] (book) ............................................................... 193
1.11.14 Don't Go Dark (blog entry) ............................................................................................................. 193
1.11.15 Reproducibility of Results in the ACM Digital Library (website) ..................................................... 193
1.11.15.1 Challenges with Reproducibility (paper) ......................................................................................... 193
1.11.15.2 Editorial: ACM TOMS Replicated Computational Results Initiative (paper) ............................... 193
1.11.15.3 Proceedings of the Reproducibility Workshop 2017 (conference) ................................................. 193
1.11.16 How to Write a Literature Review Paper? (paper) ............................................................................ 194
1.11.17 How Bibliographies, Citations, etc. are Handled in Computer Science (note) ..................................... 194
1.11.18 Greg Wilson - What We Actually Know About Software Development, and Why We Believe It’s True (talk) ........................................................................................................... 194
1.11.18.1 KOLL-2016-11-25 - Views on external and internal validity in software engineering (Janet Siegmund) (talk) .......................................................................................................... 195
1.11.18.2 NYT E03 (2016) - Controlled Experiments (Siegmund) (talk) ...................................................... 196
CS 9622 - Fall 2017 Home

Welcome to CS9622a, the grad course on non-functional software specifications (with particular reference to safety, accessibility, and sustainability)

Recent space activity

- Robert Webber
  - First Class: 10 Sept 2018 updated 3 minutes ago • view change
  - CS9622 Fall 2018 Version of Course updated yesterday at 4:54 PM • view change
  - Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability updated yesterday at 3:15 PM • view change
  - Fourteenth Lecture -- In Progress updated Jul 29, 2018 • view change
  - WCAG 2.0 - Perceivable (talk) created Jul 29, 2018

Space contributors

- Robert Webber (3 minutes ago)
- Jeff Shantz (363 days ago)

Recent Questions

How do I access the questions area?

- 1 answer • Robert Webber • Aug 02, 2017 • Space: CS 9622 - Fall 2017

@@@ README @@@

Hi. This wiki is run on an installation of software from Atlassian https://en.wikipedia.org/wiki/Atlassian running at the University of Western Ontario http://www.uwo.ca/ . This portion of the wiki system has been configured to be publically viewable. However, to comment, edit, and add to the wiki, you need to log in (which basically means you need to have an account at UWO and be associated with the course this wiki was created to support: Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability). To log in, select login from the menu that appears when you click on the person icon in the upper right hand corner of the Confluence page.

On the left hand margin, you should find under Page Tree, the roots of different sections of the course material. When you open a page, if that page is the root of a sub tree, then its children appear underneath its entry on the left margin. You can just move from node to node in the structure in the left margin, or you can look at the root node (and choose from the selections those nodes offer). The root node of each section contains a list of all the resources in that section grouped by why they are there. A good place to start is Course Overview, which explains the structure and intent of the wiki as well as provides links to the section roots. Alternately, along the top bar of the page, a bit to the left of the person icon in the right hand corner, is a search area where you can type a string and get back pages that contain that string. For example, if you type in android, you get a list of pages referencing the Android phone system on this wiki.

For those interested in taking the course, the course outline is a good place to check out Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability. General communications with students during the semester will be posted to Course Announcements.

Note that new material is constantly being added to this wiki and occasionally existing material is being modified.
Items tagged talk are generally youtube or vimeo recorded talks that are publically viewable. Items tagged paper are generally published papers whose access is controlled by the publisher (some are publically viewable, but most are not and require access to digital library subscriptions such as the ones maintained by UWO's library system https://www.lib.uwo.ca/offcampus/remote.html. In the case of items tagged book, some are publically available, some are available through campus digital subscriptions, some are in the campus libraries, and some would need to be purchased in the usual fashion for books. Items tagged website or blog are publically accessible.

Items tagged thesis (such as uwo thesis) are usually publically accessible. Many Computer Science departments made theses and other reports publically available via ftp in the 70s before web browsers. Now it has become common for universities (such as UWO) to make their entire thesis collections publically available. Unlike journal articles and blogs, theses are book sized objects (often more than 200 pages) focussed on a particular topic. In the best case, you get a good introduction to the topic and an extensive worked example. In the worst case, at least you get a bibliography of related material that can be investigated. Unlike journal articles, theses are targetted toward a more general audience as usually at least one of the examiners is not a member of the department where the thesis was written.

Course Announcements

Fall 2018 Course Overview (same as Fall 2017)

Fall 2018 Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability

The current time in New York City (which is the same as the current time in London, Ontario): https://www.timeanddate.com/worldclock/usa/new-york

Blank text board: http://www.csd.uwo.ca/~webber/CS9622/blank.html

Pastebin: https://pastebin.com/

Music for a Quiet Day

Lectures to an Empty Classroom – 2018 edition

CS9622 Fall 2017 version of course

Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability

2018 – 2019 Academic Year

1. Course Information:
   - Course Calendar
     - 6 Sept 2018: Undergraduate courses begin
     - 10 Sept 2018: First Class Meeting
     - 24 Sept 2018: in-class status report on upcoming Proposal for in Depth work
     - 1 Oct 2018: Proposal for In Depth work due
     - 8 Oct 2018: Canadian Thanksgiving Holiday https://en.wikipedia.org/wiki/Thanksgiving (this is also Undergraduate Fall Reading week)
     - 15 Oct 2018: First Progress Report due
     - 29 Oct 2018: Second Progress Report due
     - 12 Nov 2018: Third Progress Report due
     - 3 Dec 2018: Writeup due
     - 3 Dec 2018: Presentation/Demo due (note: if necessary, presentations will continue into the following weeks until done)
     - 7 Dec 2018: last day of undergraduate Fall semester before final exams
     - 17 Dec 2018: default offering of final exam: first two hours of regularly scheduled class time and meeting place (see prof is alternative time needs to be arranged)
   - Room and Time:
     - Middlesex College – MC 320; Monday, 2:30 – 5:30 pm
   - Requirements:
     - This course does not require any particular programming skills in any particular language or on any particular operating system. Of course the more experience you have with developing computer systems, the more accessible some aspects of the course content will be.
     - The target audience for the course is graduate students who are interested in how requirements like safety, accessibility, and sustainability impact the specification, design, implementation, and testing of software systems. Safety arises in the context of medical devices as well as various automated transport systems (elevators, cars, subways, trains, planes). As we increasingly create a society where people have to interact with various automated systems, we need to be concerned about whether these systems could cause personal harm. W3C’s Web Content Accessibility Guidelines (WCAG) are legal requirements for public organization web sites in many regions (including Ontario). More generally, there is the question of how to design a computer system that is not implicitly biased in favour of one portion of the population over another. In automating the infrastructure of society, the relation
between society and the environment becomes more clear and issues of the sustainability of that relationship can be more clearly addressed.

- The course’s primary website is CS 9622 - Fall 2017 Home (https://wiki.csd.uwo.ca/display/CS96222017/CS+9622+-+Fall+2017+Home) which provides a wiki for collecting material relevant to the course discussion. (Although this is the 2018 version of this course, we are re-using the web structure that was first built for the 2017 version, which is why the name is what it is. As a fall back position, if there are problems with the wiki, information regarding the course will be provided at http://www.csd.uwo.ca/~webber/CS9622/. The material discussed in class (and the source for the exam questions) will be freely available through the wiki (in conjunction with UWO's digital library subscriptions).

- Senate regulation: Unless you have either the requisites for this course or written special permission from your Dean to enroll in it, you may be removed from this course and it will be deleted from your record. This decision may not be appealed. You will receive no adjustment to your fees in the event that you are dropped from a course for failing to have the necessary prerequisites

- note: As a computer scientist, you should recognize the above as a case of patching flawed software (in this case the registrar's online enrollment software) with ‘blame the user’ policies. Hopefully you will never do anything like that in your future undertakings.

2. Instructor information: Robert E. Webber, MC384, webber@csd.uwo.ca (include course number in subject line), office hours on course announcements page (I am usually available immediately after class).

3. Course Syllabus

- The course is focussed on nonfunctional software requirements. If one views a program as a mathematical function, then the relations between its inputs and outputs are called the functional requirements (i.e., that the program implements a function that will map particular inputs to particular outputs). While most software design focusses on how to implement such functions, often there are other requirements, such as performance, security, usability, fun, safety, accessibility, and sustainability. While students may be involved in in-depth investigations relating to any of these nonfunctional requirements, the course material aims at increasing understanding of three of them: safety (cf. https://en.wikipedia.org/wiki/Software_system_safety), accessibility (cf. https://en.wikipedia.org/wiki/Accessibility), and sustainability (e.g., https://en.wikipedia.org/wiki/Sustainable_design and https://sustainability.wiki.tum.de/Sustainable+Software).

4. Course Materials

- The course material will be accessible through the digital subscriptions of the campus library system. The material will primarily consist of published papers, but theses, technical reports, and online books will also be of interest. Much interesting material on this topic can be found on youtube and other free online video services.

- Course web page: The course web page can be found as listed under Requirements in Section 1. On the course web page are links to this course outline and a frequently updated page of announcements relating to the course. Marking is done on the assumption that people check the announcements page of the course web page on a regular basis (in particular, the day before anything is due).

5. Methods of Evaluation

- Exam: There will be an exam worth 20% of the course mark. The purpose of the exam will be to ensure that people have a basic understanding of the vocabulary of the course and the major quotations from the readings. A collection of short-answer questions (and accompanying answers) will be made available to the class that address these matters. The actual exam will be a random sampling of 50 questions from the provided collection. Two hours will be available to complete the exam in. Makeup exams will be generated from the same collection using a different random number generator.

- In Depth Study: The remaining 80% of the course mark relates to individual activities undertaken to get an in depth understanding of some aspect of the area of nonfunctional requirements. There are a number of approaches one could take to this, the two most obvious are a) to undertake a literature review of some issue in the area of nonfunctional requirements and b) to undertake a programming project with particular focus on the management of some nonfunctional requirement. Both approaches can be seen as involving four stages: 1) a proposal of what is to be undertaken (5%); 2) the undertaking (with progress reports (3 progress reports, 5% each)); 3) the writeup of the undertaking (50%); and 4) the presentation/demo of the undertaking (10%).

6. Additional Statements

- Statement on Use of Electronic Devices: The exam will be closed book, closed notes, with no electronic devices allowed, with particular reference to any electronic devices that are capable of communication and/or storing information.

- Statement on Use of Personal Response Systems: Clickers will not be used in this class as it is nearly impossible to use them and protect student privacy, as any Computer Scientist should know.

- Statement on Academic Offenses: Scholastic offenses are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site:http://www.uwo.ca/univsec/academic_policies/appeals/scholastic_discipline_undergrad.pdf

- Statement on Academic Offenses (graduate version): Scholastic offenses are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site:http://www.uwo.ca/univsec/pdf/academic_policies/appeals/scholastic_discipline_grad.pdf

- Regarding Plagiarism-Checking Software: Such software will not be used in this class as it is nearly impossible to use them and protect student privacy, as any Computer Scientist should know. In the case of both material handed in and tests and/or exams, any plagiarism checking will be done the old fashioned way, by a person looking at the material in question.

- Senate regulation: Students who are in emotional/mental distress should refer to Mental Health @ Western http://www.uwo.ca/uwocom/mentalhealth/

- Senate regulation: For UWO Policy on Accommodation for Medical Illness and a downloadable SMC see:http://www.uwo.ca/univsec/pdf/academic_policies/appeals/accommodation_medical.pdf [downloadable Student Medical Certificate (SMC): https://studentservices.uwo.ca/ under the Medical Documentation Heading] Students seeking academic accommodation on medical grounds for any missed tests, exams, participation components and/or assignments worth 10% or more of their final grade must apply to the Academic Counselling office of their home Faculty and provide documentation. Academic accommodation cannot be granted by the instructor or department.

- Department specific information
  - CS Department Rules of Ethical Conduct
  - CS Department Specific Information on Scholastic Offences

7. Specification for this document

- course outline requirements
- requirements regarding undergrad evaluation
Fall 2017 Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability

2017 – 2018 Academic Year

1. Course Information:
   - Course Calendar
     - 11 Sept 2017: First Class Meeting
     - 2 Oct 2017: Proposal for In Depth work due
     - 9 Oct 2017: Canadian Thanksgiving Holiday https://en.wikipedia.org/wiki/Thanksgiving (this is also Undergraduate Fall Reading week)
     - 16 Oct 2017: First Progress Report due
     - 13 Nov 2017: Third Progress Report due
     - 4 Dec 2017: Writeup due
     - 4 Dec 2017: Presentation/Demo due (note: if necessary, presentations will continue into the following weeks until done)
     - 8 Dec 2017: last day of undergraduate Fall semester before final exams
   - Room and Time:
     - Middlesex College – MC 300; Monday, 2:30 – 5:30
   - Requirements:
     - This course does not require any particular programming skills in any particular language or on any particular operating system. Of course the more experience you have with developing computer systems, the more accessible some aspects of the course content will be.
     - The target audience for the course is graduate students who are interested in how requirements like safety, accessibility, and sustainability impact the specification, design, implementation, and testing of software systems. Safety arises in the context of medical devices as well as various automated transport systems (elevators, cars, subways, trains, planes). As we increasingly create a society where people have to interact with various automated systems, we need to be concerned about whether these systems could cause personal harm. W3C’s Web Content Accessibility Guidelines (WCAG) are legal requirements for public organization web sites in many regions (including Ontario). More generally, there is the question of how to design a computer system that is not implicitly biased in favour of one portion of the population over another. In automating the infrastructure of society, the relation between society and the environment becomes more clear and issues of the sustainability of that relationship can be more clearly addressed.
     - The course’s primary website is CS 9622 - Fall 2017 Home (https://wiki.csd.uwo.ca/display/CS96222017/CS+9622+-+Fall+2017+Home) which provides a wiki for collecting material relevant to the course discussion. As a fall back position, if there are problems with the wiki, information regarding the course will be provided at http://www.csd.uwo.ca/~webber/CS9622/. The material discussed in class (and the source for the exam questions) will be freely available through the wiki (in conjunction with UWO's digital library subscriptions).
     - Senate regulation: Unless you have either the requisites for this course or written special permission from your Dean to enroll in it, you may be removed from this course and it will be deleted from your record. This decision may not be appealed. You will receive no adjustment to your fees in the event that you are dropped from a course for failing to have the necessary prerequisites
     - note: As a computer scientist, you should recognize the above as a case of patching flawed software (in this case the registrar’s online enrollment software) with ‘blame the user’ policies. Hopefully you will never do anything like that in your future undertakings.

2. Instructor information: Robert E. Webber, MC384, webber@csd.uwo.ca (include course number in subject line), office hours on course announcements page (I am usually available immediately after class).

3. Course Syllabus
   - The course is interested in nonfunctional software requirements. If one views a program as a mathematical function, then the relations between its inputs and outputs are called the functional requirements (i.e., that the program implements a function that will map particular inputs to particular outputs). While most software design focusses on how to implement such functions, often there are other requirements, such as performance, security, usability, fun, safety, accessibility, and sustainability. While students may be involved in in-depth investigations relating to any of these nonfunctional requirements, the course material aims at increasing understanding of three of them: safety (cf. https://en.wikipedia.org/wiki/Software_system_safety), accessibility (cf. https://en.wikipedia.org/wiki/Accessibility), and sustainability (e.g., https://en.wikipedia.org/wiki/Sustainable_design and https://sustainability.wiki.tum.de/Sustainable+Software).

4. Course Materials
   - The course material will be accessible through the digital subscriptions of the campus library system. It will primarily consist of published papers, but also video talks of interest.
   - Course web page: The course web page can be found ?. On it are links to this course outline and a frequently updated page of announcements relating to the course. Marking is done on the assumption that people check the announcements page of the course web page on a regular basis (in particular, the day before anything is due). Once Allianz accounts have been set up for everyone who added the class by the add date, the course announcements will move to ?

5. Methods of Evaluation
   - Exam: There will be an exam worth 20% of the course mark. The purpose of the exam will be to ensure that people have a basic understanding of the vocabulary of the course and the major quotations from the readings. A collection of short-answer questions (and accompanying answers) will be made available to the class that address these matters. The actual exam will be a random sampling of 50 questions from the provided collection. Two hours will be available to complete the exam in. Makeup exams will be generated from the same collection using a different random number generator.
In Depth Study: The remaining 80% of the course mark relates to individual activities undertaken to get an in-depth understanding of some aspect of the area of nonfunctional requirements. There are a number of approaches one could take to this, the two most obvious are a) to undertake a literature review of some issue in the area of nonfunctional requirements and b) to undertake a programming project with particular focus on the management of some nonfunctional requirement. Both approaches can be seen as involving four stages: 1) a proposal of what is to be undertaken (5%); 2) the undertaking (with progress reports (3 progress reports, 5% each)); 3) the writeup of the undertaking (50%); and 4) the presentation/demo of the undertaking (10%).

6. Additional Statements

- **Statement on Use of Electronic Devices:** The exam will be closed book, closed notes, with no electronic devices allowed, with particular reference to any electronic devices that are capable of communication and/or storing information.
- **Statement on Use of Personal Response Systems:** Clickers will not be used in this class as it is nearly impossible to use them and protect student privacy, as any Computer Scientist should know.
- **Statement on Academic Offenses:** Scholastic offenses are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site: http://www.uwo.ca/univsec/academic_policies/appeals/scholastic_discipline_undergrad.pdf
- **Statement on Academic Offenses (graduate version):** Scholastic offenses are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site: http://www.uwo.ca/univsec/pdf/academic_policies/appeals/scholastic_discipline_grad.pdf
- **Regarding Plagiarism-Checking Software:** Such software will not be used in this class as it is nearly impossible to use them and protect student privacy, as any Computer Scientist should know. In the case of both material handed in and tests and/or exams, any plagiarism checking will be done the old fashioned way, by a person looking at the material in question.
- **Senate regulation:** Students who are in emotional/mental distress should refer to Mental Health @ Western http://www.uwo.ca/uwocom/mentalhealth/.

7. Specification for this document

- course outline requirements
- requirements regarding undergrad evaluation
- requirements regarding scheduling of tasks

**Course Overview**

In Computer Science, we often think about programs as ends in themselves. We ask the question of how to write a payroll program (cf Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018) ), often without giving much thought to the bigger question of how the pay system works and how we in tend to change it. While the payroll program may have many interacting parts, it is just a single component in the larger pay system. Ultimately the usefulness of the payroll program is derived from the success of the pay system in which it is embedded. While the need to change the payroll program sometimes comes from internal component problems, more often it comes from difficulties derived from its interaction with the rest of the pay system. Properties of the payroll program, such as security, derive from how it used in the wider pay system and are not internal properties of the software itself. A systems person would say that security is an emergent property of the payroll system itself and not a property of any particular component of the system.

The interface between the pay systems designer and the payroll program designer is the requirements (constraints) that the payroll program is supposed to satisfy in order to function properly in the pay system. Functional requirements are about what the program is supposed to do (such as withhold taxes properly). Nonfunctional requirements are usually about emergent properties of the system that the program is supposed to fit into (such as being `secure`).

This particular course is interested in three particular nonfunctional requirements: safety, accessibility, and sustainability. Each of these clearly raises issues of how software components fit into larger socio-technical systems. Some of the basic questions that arise are what is meant by each of these requirements and how do you verify that you have actually met such requirements.

The wiki built of material to support this course has many parts. Six parts can be thought of as context components, i.e., providing information more general than the task at hand to provide background for the main goal of the course:

- Thinking about systems
- Understanding humans
- Overview of non-functional software requirements and requirements engineering
- From requirements to design
- How other non-functional requirements figure into software systems
- Writing papers in computer science

The material on the three focus topics of the course can be found in:

- How safety requirements figure into software systems
- How accessibility requirements figure into software systems
- How sustainability requirements figure into software systems
Eighth class (Monday) 6 Nov 2017

music for this Monday:
- Tim Ovens plays John Cage · Sonata X for Prepared Piano (4:03) https://www.youtube.com/watch?v=ce4TCh0gGM
- Ants Marching/Ode To Joy - 4 Guys, 3 min, 2 cellos, 1 piano - The Piano Guys (3:49) https://www.youtube.com/watch?v=17GLE-16_3g

music from last Monday Seventh class (Monday) 30 Oct 2017

followup power failure from last Monday Seventh class (Monday) 30 Oct 2017

followup: four classic object-oriented programming books
- Design Patterns: Elements of Reusable Object-Oriented Software (book) https://en.wikipedia.org/wiki/Design_Patterns
- Refactoring: Improving the Design of Existing Code (book)– there is a Ruby specific version also https://en.wikipedia.org/wiki/Martin_Fowler

followup: autonomic computing https://en.wikipedia.org/wiki/Autonomic_computing
- Autonomic Computing for DB2 (IBM) 10:28 https://www.youtube.com/watch?v=VCOSdlkdcUc

a look at the remaining schedule: Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability
- 13 Nov 2017 third progress report due
- 20 Nov 2017 3 minute practice talks
- 27 Nov 2017
- 4 Dec 2017 writeup due, presentation due (roughly 10 presentations 10 – 15 minutes each tightly fills time available)
- how to handle final exam?

readings:
- Evaluating oral presentations (paper) – note the author references Writing for Computer Science [electronic resource] / by Justin Zobel (book) which is online thru the digital library link and includes the coverage of presentations on pages 237 to 254. Writing about speaking is sometimes awkward – youtube, of course, gives us many examples of people speaking:
  - How to open and close presentations? - Presentation lesson from Mark Powell (7:36) https://www.youtube.com/watch?v=Yl_FJAOcFgQ
  - example presentation: Literature Review Presentation (13:41 presentation of literature review on game playing in education) https://www.youtube.com/watch?v=0_3VG6i-SGs – slides were crowded, but otherwise good example of what can be done with this sort of presentation.
• One can think of a 10 page paper as a 50 minute presentation (taking about 5 minutes to read a page worth of text). A more standard 10 to 15 minute talk is a compressed version of the paper with more narrow focus. And inside each 10 to 15 minute talk, there is a 3 minute talk (Learn how to give a 3 minute presentation in under 3 minutes (2:42) https://www.youtube.com/watch?v=54BPsCtllNg and Winning Tips for Preparing a Successful Three-Minute Thesis 3MT® Presentation (12:32) https://www.youtube.com/watch?v=D_1MK3UUbIQM). A popular university competition is the 3 minute thesis presentation, e.g.,

• 3 Minute Thesis 2017 - Michael Miljanovic (3:09 UOIT) https://www.youtube.com/watch?v=f9ftNhYyL8
• Three Minute Thesis (3MT) - Jenna Butler - 1st Place 2015 (3:00 CS UWO) https://www.youtube.com/watch?v=GyZd7tcrKWLg&index=57&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb – no slides
• Dapping Gao - 3MT 2017 (2:57 UWO) https://www.youtube.com/watch?v=fd_mFZoxKmI&index=8&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb
• Yoah Sui: Standing Up for Student Health - 3MT (2:44 UWO) https://www.youtube.com/watch?v=Z9thAPkmQzc&index=35&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb
• and again Yoah Sui - 3MT 2017 (2:50 UWO) https://www.youtube.com/watch?v=z2P3f9QOWZB&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb&index=5
• Three Minute Thesis (3MT) - Ryan Armstrong - 2nd Place 2015 (3:15 UWO) https://www.youtube.com/watch?v=1X0DKcHyKo&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb&index=40 – no slides
• and again Three Minute Thesis (3MT): Ryan Armstrong - 2016 (3:11 UWO augmented reality for surgeons) http://www.youtube.com/watch?v=F40DKcHyKo&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb
• Nicolette Noonan: Statistical Language Learning - 3MT (3:13 UWO) https://www.youtube.com/watch?v=N9Qh5JwEIjo&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb
• and again Nicolette Noonan: Statistical Language Learning - 3MT (3:13 UWO) https://www.youtube.com/watch?v=4nI0DKcHyKo&list=PLbZM4TW_GWGT_nvZ_SDdpv8KpCQqqGfb

• Goal-Based Requirements Analysis (paper)
• Privacy Impacts of IoT Devices: A SmartTV Case Study (paper) – note: this paper uses approach outlined in above paper Goal-Based Requirements Analysis (paper)
• Engineering a safer world: systems thinking applied to safety (book) , specifically http://sunnyday.mit.edu/safer-world.pdf Section 8 pp 179 – 196 (upto and including Subsection 8.5)

with usual comments Notes on readings for eighth class

Notes on readings for eighth class

• Goal-Based Requirements Analysis (paper)
  • determining requirements by first investigating goals
  • looks closer at how to find goals
• Privacy Impacts of IoT Devices: A SmartTV Case Study (paper) – note: this paper uses approach outlined in above paper Goal-Based Requirements Analysis (paper)
  • determining system goals by looking at privacy policy enumerates goals
  • relating privacy policy goals to conformance to privacy policy regulations – particularly idea of consent
• Engineering a safer world: systems thinking applied to safety (book) , specifically http://sunnyday.mit.edu/safer-world.pdf Section 8 pp 179 – 196 (upto and including Subsection 8.5)
  • looking for ways the system could fail
  • human components more complicated than other components – automation generally requires extra training

Eleventh class (Monday) 27 Nov 2017

Music:

• Bach Fugue 46 by Emmy https://www.youtube.com/watch?v=S9ghZMgjxbs 4:36 (Emmy Howell Al by David H Cope, 2012)
• Attention RNN Song Sample 1 https://ai.google/products/attention-rnn-lookback-rnn/ 1:23 (14 July 2016 – Google AI)
• DeepBach: harmonization in the style of Bach generated using deep learning 0:59 (Sony CSL) http://www.flow-machines.com/deepbach-polyphonic-music-generation-bach-chorales/
• Alex Da Kid - Not Easy (Official Video) ft. X Ambassadors, Elle King, Wiz Khalifa https://www.youtube.com/watch?v=u-e90ELRnnQ 4:06

Notes:

• http://en.wikipedia.org/wiki/Signalling_System_No._7
• As you are preparing your final writeup, it is worthwhile thinking about how it looks to a reader – in particular, your goal is to make it easy for the reader to figure out the key aspects of your writing. Recall from class 2 Second Class (Monday) 18 Sept 2017 the advice for readers (as well as its expansion in Third Class (Monday) 25 Sept 2017):
  • What is the main point the author/speaker is trying to get across?
  • Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into what else you know about this topic?
  • Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new vocabulary help organize the presentation?
  • Can you see how this information might be used by you in some future undertaking?
• As you are preparing your final writeup, it is also worthwhile remembering the discussion of The Rationale for Writing an LRP and Added Value in Sixth Class (Monday) 23 Oct 2017. In Notes re readings for sixth class, there were some writing comments related to Engineeri ng a safer world: systems thinking applied to safety (book) that are also worth considering as to how they would apply to your writing. Also don’t forget the https://grammark.org/dist/# and https://languagetool.org/ tools mentioned in notes re readings for 13 Nov 2017.
• Toward the final exam has been updated to include links related to questions for the first 10 classes (roughly 80).

Class Material:

• Predictive human performance modeling made easy (paper)
  • https://en.wikipedia.org/wiki/GOMS
  • https://en.wikipedia.org/wiki/Fitts%27s_law
• ACT: A simple theory of complex cognition (paper)
  • ACT-R system home page (website)
  • Understanding humans (new top level section)
• Deploying CogTool: integrating quantitative usability assessment into real-world software development (paper)
  • https://cogtool.wordpress.com/about/
  • https://github.com/cogtool
• Measuring the security impacts of password policies using cognitive behavioral agent-based modeling (paper)
  • https://www.isi.edu/~blythe/Dash/
  • note: itemized lists, figures, and tables would have made section 3.2 much easier to read.

Fifth Class Monday 16 Oct 2017


Email was sent out when the proposals were handed in letting people know about:

• Format for Progress Reports

One issue that many people missed on their proposals was a discussion of how time resources were to be managed for the project (which would usually involve some sort of time line with estimates of how much time the various tasks on the time line would take). This sort of thing was alluded to in Third Class (Monday) 25 Sept 2017 at the bottom of the page under objectives regarding the importance of resources. Interestingly, if we look at the book Thesis Projects A Guide for Students in Computer Science and Information Systems (book) that was the basis for the third class discussion of proposals, we find that the ‘time plan’ is not mentioned in proposals, but rather discussed in the subsequent chapter on Developing you Aim https://link.springer.com/chapter/10.1007/978-1-84800-009-4_7 (which is one of the parts of the proposal). Incidentally, the time plan gets a brief mention again in the 2 page Chapter 9 Following The Objectives https://link.springer.com/chapter/10.1007/978-1-84800-009-4_9. And, as was brilliantly said in Third Class (Monday) 25 Sept 2017: specific goals for each progress report are needed in order to be able to monitor progress and know when things need to be rethought.

Also, perhaps of interest in understanding the roles of proposal and progress report is to consider Chapter 3 Actors Involved, their Roles and Relationships https://link.springer.com/chapter/10.1007/978-1-84800-009-4_9. In thesis supervision, it is traditional to exchange information among the actors via conversations and regular meetings. As organizations get larger, such exchanges become written documents. A classroom where many people are working on different tasks is usually organized more like a large organization than a thesis supervision.

Many people have trouble with time estimates. RubyConf 2015 - I Estimate this Talk will be 20 Minutes Long, Give or Take 10 Minutes (talk) and RailsConf 2014 - Unreasonable Estimates and Improbable Goals by Adam Sanderson (talk)

followup: Oct 4 government report on Phoenix Pay released as well as note on conflict of interest in original project proposal Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018)

followup: from Third Class (Monday) 25 Sept 2017. The paper Practical Keystroke Timing Attacks in Sandboxed JavaScript. (paper) shows how a small amount of Javascript code running in an ad could be monitoring timing information on mobile devices and what sort of information it could infer from that information. This is a 2017 paper and so also discusses the work on timing attacks done since the paper we looked at on the 25th.
In depth topics in proposals:

- RSA cryptography for cloud security
- Source code-based security testing
- Non-functional requirements for Big Data (security/privacy)
- Security requirements of cloud data
- Sustainability and security issues in emergency services
- Relation between sustainability and maintainability
- Green software and green software engineering
- Development of personal productivity web app with focus on accessibility issues
- Accessibility issues in recommender systems
- Usage of UML for safety analysis

Nearly half of the topics were well focussed with the remainder needing further work to be able to say something interesting in just 10 pages. As was brilliantly said in Third Class (Monday) 25 Sept 2017: you should pick something that you think would be interesting/useful to you and that you can explain why that is, the more focussed the better.

Directions for the next few week’s readings:

- Sustainability and failure handling in critical systems
- Quantitative measures of nonfunctional requirements – in particular, sustainability
- Sustainability in terms of maintainability – can maintainability be measured?
- Sustainability in software engineering
- Sustainability in software engineering process and green software
- Security issues in the cloud
- Accessibility studies on recommender systems
- Software testing
- Operating system safety

Hmmm. In distributed systems (and operating systems?) there are ‘safety’ properties https://en.wikipedia.org/wiki/Safety_property (that specified bad things won’t happen) – this is not the usage of the term “safety” here in that it isn’t a nonfunctional requirement, but instead it is an aspect of the formal correctness of the system.

Perhaps a good place to address these issues would be with:

- Requirements engineering: a roadmap (paper) – google scholar lists 2257 citations for this paper, so it certainly had influence (acm lists 253 citations, but that too is much larger than usual)
- Measuring software sustainability (paper) – maintenance view
- A systematic literature review for software sustainability measures (paper) – Green view
- Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities (U Pittsburgh Thesis) – we would be specifically interested in: Chapter 3: Constructing Measurement Metrics For Web Accessibility Evaluation
- Measuring the user experience on a large scale: user-centered metrics for web applications (paper)

usual comments about advisability of reading (looking at) the papers first:

- Notes on readings for fifth class

Notes on readings for fifth class

(usual comments about the advisability of reading the papers first)

- Requirements engineering: a roadmap (paper) – google scholar lists 2257 citations for this paper, so it certainly had influence (acm lists 253 citations, but that too is much larger than usual)
  - the first paragraph of the intro gives a nice definition of ‘requirements engineering’ and later in the section there is a good list of the 5 core activities.
  - although focussed on directions for the future, much of this article is a readable explanation of the state of the art.
  - For us, interesting that one of the key research pointers in this look toward the future of requirements engineering is Development of richer models for capturing and analysing non-functional requirements. See header for article and Section 10 roadmap.
  - Section 10 gives us a pointer to reference 50: which is in the same conference: Maibaum, T. S. E. (2000). “Mathematical Foundations of Software Engineering: A Roadmap. https://dl.acm.org/citation.cfm?id=336512.336548 In his section on Criteria and Specifications, he doesn't use the term 'non-functional requirements', but instead speaks about 'qualitative goals'. He takes the ‘scientific’ view that things get better when we convert qualitative goals to quantitative goals.
  - Section 5.5 of this article takes a similar view with the title: Modelling Non-Functional Requirements (NFRs)
  - Section 5.5 gives us 6 relevant references
    - 69 Suzanne Robertson , James Robertson, Mastering the requirements process, ACM Press/Addison-Wesley Publishing Co., New York, NY, 1999


• Measuring software sustainability (paper) – maintenance view
  • maintainability is about the software whereas sustainability is about the whole software evolution process including employees and customers.
  • focus on the sustainment team and organization within which the team operates as well as documentation and other software artefacts that are part of the software evolution process
  • also concern with the operational domain -- the environment where the software runs, the customers and users, and the evolutionary requirements
  • models sustainment after IEEE Software Maintenance Standard as distinct from `mere bug fixes'
  • re measuring security, see Alberts 2002 book on OCTAVE approach
  • distinguish adaptive, corrective, and perfective changes
  • count how many of each change type occurs and how quickly it is handled and merge in a weighted average WMRD.
  • other causal measures are used to guide the effort of improving the WMRD.
  • fundamental observation is that we can measure a complicated process by focussing on the time resources it consumes across its lifespan (as opposed to the time it takes to perform a small subtask).

• A systematic literature review for software sustainability measures (paper) – Green view
  • quotes following green definition: Sustainable Software is software, whose direct and indirect negative impacts on economy, society, human beings, and environment that result from development, deployment, and usage of the software are minimal and/or which have a positive effect on sustainable development
  • sadly relies on abstracts being well written
  • their searches found 740 papers. reading abstracts and titles they selected 70. reading these full papers, they ended up with 16 papers.
  • these 16 papers listed in Appendix 1.
  • note: this paper was published 2013. how would we figure out if there had been progress since then?
  • their search went back to 1992, but the oldest paper they picked was 2003.

• Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities (U Pittsburgh Thesis) – we would be specifically interested in: Chapter 3: Constructing Measurement Metrics For Web Accessibility Evaluation
  • WCAG gives us 91 checkpoints grouped in 3 levels of priority
  • 25 of the checkpoints can be automatically verified and applied to just the main page of a web site to get a sense of how compliant the web is.
  • 3.2.4 want to measure relative accessibility
  • also want to be able to track rate of change in accessibility
  • metric should be normalized for size and complexity of web site
  • metric should be efficient to compute
  • automated scoring gives appearance of objectivity
  • 3.3 Web Accessibility Barrier
  • weight by priority the 25 automatically measurable violations that occur and divide by number of objects that could have that violation on a single web page. then average the measure on each web page against the number of web pages.
  • complexity here means number of objects on a web page.

• Measuring the user experience on a large scale: user-centered metrics for web applications (paper)
  • web applications let us use web server logs to track product usage patterns to look for indications of usability
  • A/B testing (not all users of a web site see the same web site)
  • quote: PULSE metrics: Page views, Uptime, Latency, Seven-day active users (i.e. the number of unique users who used the product at least once in the last week), and Earnings.
  • proposed to complement with
  • quote: HEART: Happiness, Engagement, Adoption, Retention, and Task success
  • use in-product surveys to measure Happiness
  • Engagement derived from direct measures in the style of usage per user per week
  • Adoption tracks how many new users per time period
  • Retention tracks how many returning users per time period
  • Task success is tricky requiring automatic determination from log data of what a user is trying to do and how many errors they are encountering during the attempt or if they succeed at all.
  • to get people to use framework, 3 step process:
    1) determine goal of product
    2) determine signals in user behavior or attitudes that indicate success
    3) create metric to measure signals
  • 2010 paper
  • used on 20 google products. did this have a future?

RailsConf 2014 - Unreasonable Estimates and Improbable Goals by Adam Sanderson (talk)

https://www.youtube.com/watch?v=b6Q2dQZ6GIY 28:43

RubyConf 2015 - I Estimate this Talk will be 20 Minutes Long, Give or Take 10 Minutes (talk)
First Class (Monday) 11 Sept 2017

I plan to visit and discuss the following wiki pages:

- **Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability**
- NOTE: Final Exam date was not in the course outline. The response was that it would probably be 4 Dec if there was room for it and the presentations that day and, if not, it would be the week before.
- **@@README @@@**
- **Course Overview**
- **Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018)**
- The wiki built of material to support this course has many parts. Four parts can be thought of as context components, i.e., providing information more general than the task at hand to provide background for the main goal of the course:
  - Thinking about systems
  - Overview of non-functional software requirements and requirements engineering
  - How other non-functional requirements figure into software systems
  - Writing papers in computer science
- The material on the three focus topics of the course can be found in:
  - How safety requirements figure into software systems
  - How accessibility requirements figure into software systems
  - How sustainability requirements figure into software systems

Some introductory material to look at to lay the groundwork for the next class discussion:

- on systems: In A World of Systems (video) (9 minute video)
- on requirements: An introduction to Requirements Engineering (talk) (10 minute talk)
- on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
- on accessibility: Universal Design (talk) (24 minute talk)
- on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)

**Format for Progress Reports**

Following the class discussion **Third Class (Monday) 25 Sept 2017**, your proposal should have had an objectives section and that section should have shown how you expect to allocate the 60 hours of time the project is scheduled to take across the semester starting from the proposal passing through three roughly evenly spaced progress reports and then concluding with the final writeup and presentation. Like the proposal, the progress report should be one or two pages at most (not counting optional attached writings). The progress report should thus have the following format:

- Your Name
- Your UWO email address
- A title that is Progress Report Number for Title of Your Project From Proposal
- Date
- and then the body of the progress report as described below.

The body of the progress report then has the following sections:

- **Previous Timeline**
  - note: for first progress report, this would come from your proposal. after that it would be the version of timeline as of the most recent progress report
- **Progress Made Against Timeline**
  - note: summary of issues that arose while doing what you were expecting to do for this time period. should state what was done as well as what difficulties were encountered.
- **Proposed New Timeline (optional)**
  - note: if either you have fallen behind the expected progress indicated by the previous timeline or you want to change the goal of your project; here is where you make your adjustments and state your new aim and new timeline.
- **Writings (optional)**
  - note: as mentioned in **Third Class (Monday) 25 Sept 2017** it is generally a good idea to be writing as you go along rather than trying to do it all at the end of the project. if you have written new material during the time frame of this report and want feedback on it, this would be where you would mention it and attach it to your progress report.

Handin options for progress reports are the same as those given for the proposal.
Fourth Class (Monday, 2 Oct 2017)

Music:
- Claude Debussy - Prelude to the Afternoon of a Faun https://www.youtube.com/watch?v=9_7loz-HWUM
- Clair de Lune - Jeremy Denk and Melanie Torres - 11/28/2015 https://www.youtube.com/watch?v=Wn30KZU0bKA

Readings for class:
- A Philosophical Look at System Dynamics (talk) – 54 minutes
- Integration between requirements engineering and safety analysis: A systematic literature review (paper) – 25 pages
  - Note: this paper is a bit long, but I have chosen it to give an example of what systematic literature reviews look like (as it seems many people are forming proposals in that direction).
  - It is also worth noting that this is a rare example of a paper that has a very useful abstract.
- Making the web easier to see with opportunistic accessibility improvement (paper) – 6 pages
- Performativity in sustainable interaction: the case of seasonal grocery shopping in ecofriends (paper) – 10 pages

As usual, save the notes until after you have had a chance to consider the readings themselves as per comments from previous two classes. Notes on readings for Fourth Class (Monday, 2 Oct 2017)

Note:
- we should do a check on what topics people want the class readings to focus on similar to what we did two weeks ago.
- proposal write ups due today
- no class next week as it is Canadian Thanksgiving holiday.

Notes on readings for Fourth Class (Monday, 2 Oct 2017)

- A Philosophical Look at System Dynamics (talk)
  - the overall goal is to present the fundamentals of how system dynamics is used as an approach to learn about complex systems at a nontechnical level
  - dynamic models are interested in how systems ‘change over time’.
  - references Lazlo as the approach to how systems are defined. [https://en.wikipedia.org/wiki/Ervin_L%C3%A1szl%C3%B3 and http://ervinlaszlo.com/]. see also The Contribution of the Systems Sciences to the Humanities (paper) by Lazlo and son.
  - four important aspects of system dynamics
    - causal linkages (starting 3:09)
      - start by drawing arrows about what is connected to what by relationships we think are causal (we believe there is an underlying mechanism that causes one component to affect another).
      - a plus sign on the arrow indicates more of the originating component makes more of the terminating component, minus sign used analogously.
      - 5:48 example graph of doctors per person versus food per person. illustrating difference between correlated and causally related.
      - 8:39 example graph of life expectancy versus calories available in diet. a nonlinear relation. same with health services versus life expectancy
      - most systems relationships are nonlinear
      - 12:00 feedback loop example in modelling food and health system. uses squiggle on arrow to indicate nonlinear relations.
      - 14:53 in addition to nonlinearities, look for delays (indicated by box on tail of arrow). which delays matter depends on time frame of system analysis.
    - feedback loops (starting 18:00) closed chains of causal relations.
      - open system view: there is an input and an output
      - instead view world as a state of a system with actions that change the state of the system and the state of the system changes future decisions on how system will be changing creating a loop between state and actions.
      - 21:09 keg of beer, empty glass system example. a goal with a desired level.
      - distinction between positive and negative feedback loops. an even number of negative signs creates a positive loop and an odd number of negative signs creates a negative loop.
      - 24:00 bank deposit, interest example.
      - 26:15 multiloop model example oil company
      - whenever you hear someone say that A causes B, you should look for ways that B influences A creating a loop.
      - note: these loops mean external events don’t drive the systems models nor does time. instead it is the state of the system that determines how it will change and so these models are called state determined systems – so when there are problems, you look inside the system rather than outside it.
      - rates and levels (starting with beer example 21:09)
        - 37:43 in systems dynamics, the state of the system is called a level and the actions that can be applied to the system are called a rate.
in some systems diagrams, a box is used to represent a level and a valve symbol is used to represent a rate. a solid arrow for what is going into the system state and a dotted line for what is coming from the system state to the level.

41:00 revisits beer example in new notation and bank balance example as well. gives a start on redoing the oil example.

in the oil example, a circle is used to represent an auxiliary that basically represents a separate equation. mentions dynamo diagrams. https://en.wikipedia.org/wiki/DYNAMO_(programming_language)

- structural behavioral relationships (starting 48:18)

- after building a model, we gain a lot of knowledge about how the parts of the system work together.
- we expect positive feedback loops in isolation to generate exponential growth or decline.
- a negative feedback loop drives the system to a goal level with diminishing returns
- a positive loop connect to a negative loop gives a sigmoid function https://en.wikipedia.org/wiki/Sigmoid_function, delays introduce wiggles into the sigmoid.
- thus from data measuring behavior of a system, one can guess at the structure of the underlining model.

- Integration between requirements engineering and safety analysis: A systematic literature review (paper)

- https://en.wikipedia.org/wiki/Systematic_review quote: Systematic reviews are a type of literature review that collects and critically analyzes multiple research studies or papers, using methods that are selected before one or more research questions are formulated, and then finding and analyzing studies that relate to and answer those questions in a structured methodology.[1]
They are designed to provide a complete, exhaustive summary of current literature relevant to a research question. Systematic reviews of randomized controlled trials are key in the practice of evidence-based medicine,[2] and a review of existing studies is often quicker and cheaper than embarking on a new study.

- https://en.wikipedia.org/wiki/Systematic_review#Stages see also steps for systematic review process in Figure 1 of this paper https://ars.els-cdn.com/content/1-s2.0-S0164121216302333-gr1_lrg.jpg

- https://en.wikipedia.org/wiki/Literature_review which cited

- getting back to the paper:

- when we look at the conclusions, we get seven recommendations, but nothing that traces them back to the paper. we do get some interesting comments worth looking into:

  that only 57 of 1037 papers were used. raises the question of how they were chosen.
  in section 3.2 on search strategy we are given info on which literature databases they searched and what queries they used. also references to standard approaches to doing this PICOC and SART. section 3.3 discusses inclusion and exclusion criteria. figure 3 https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr2_lrg.jpg shows us how the 57 were chosen.
  the table 3 Extraction Form under section 3.5 on data extraction and synthesis shows what questions they went into each paper and tried to answer.
  section 3.6 (quality assessment) then mentions 20 questions that were used to assess the quality of the 57 papers with regards to this study.

- under need for tools said that 27 tools were mentioned. which tools were they?

  - table 8 shows techniques that were mentioned in papers http://www.sciencedirect.com/science/article/pii/S0164121216302333#tbl0008
  - tools used were listed in section 4.1 as this was RQ1.4's question. https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr10_lrg.jpg
  - it was common for a tool just to be mentioned in the paper proposing it. 18 existing tools got mentioned in only one paper. Sparx System Enterprise Architect was mentioned in 4 papers. https://en.wikipedia.org/wiki/Enterprise_Architect_(software) and http://sparxsystems.com/ (their site provides a UML tutorial http://sparxsystems.com.uml-tutorial.html so they are probably based in that methodology).

  as mentioned before, this paper has a nice abstract (and even a highlights section) of the five highlight points, two mention taxonomies, yet the prefix taxonomy does not appear in the conclusions. it does appear 32 times in the paper https://en.wikipedia.org/wiki/Taxonomy and https://en.wikipedia.org/wiki/Safety_taxonomy

  - twice in the highlights
  - once in the results section of abstract. it kind of looks like these taxonomies (4 of them) are the key contribution of the paper.
  - once in introduction
  - in related work there two mentions of taxonomies in separate paragraphs.
  - in section 3.1 on research question RQ1.2 and RQ1.3’s description includes reference to being used to build taxonomies.
  - in section 3.5 on data extraction and synthesis it mentions that synthesis involved creating a common vocabulary across articles to build the taxonomies (1 mention),

  - Figure 6 gives us the first taxonomy on general techniques for safety analysis https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr6_lrg.jpg
  - Figure 7 gives us a taxonomy on general techniques used for hazard analysis https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr7_lrg.jpg
  - Figure 8 gives us a taxonomy of safety information https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr8_lrg.jpg
  - Figure 9 gives us a taxonomy on hazard information (the most complicated of the taxonomies) https://ars.els-cdn.com/content/image/1-s2.0-S0164121216302333-gr9_lrg.jpg

  - the stem taxonomy appears twice in the bibliography. once in M. Unterkalmeister, R. Feldt, T. Gorschek A taxonomy for requirements engineering and safety engineering software test alignment ACM Trans. Softw. Eng. Methodol. (TOSEM), 23 (2) (2014), p. 16 whose last author is an author on this paper as well.
• the paper contains a number of comments on why these taxonomies are useful to the progress of the field which can be found by searching for taxonomies

- Making the web easier to see with opportunistic accessibility improvement (paper)
  - 30 second video by author: Opportunistic Accessibility Improvement https://www.youtube.com/watch?v=BvYuo67zz0M
  - oppaccess.js doesn't seem to be available, see https://twitter.com/jeffbigham/status/866730295797612544
  - the paper does include some discussion of how they went about implementing this code
  - key vocabulary introduced by author is notion of opportunistic accessibility.
  - opportunistic appears 13 times in paper where opportunistic accessibility appears only 6 times indicating pdf search has problems with phrases across multiple lines and probably hyphenations as well.

- Performativity in sustainable interaction: the case of seasonal grocery shopping in ecofriends (paper)
  - 30 second video introducing Performativity in Sustainable Interaction: The Case of Seasonal Grocery Shopping in EcoFriends https://www.youtube.com/watch?time_continue=3&v=NMP4LD5G4iM
  - looking at the references and the intro we see this sets in the Sustainable HCI camp.
  - from title to conclusions, performative design as an approach to sustainable HCI is the theme here. https://en.wikipedia.org/wiki/Performativity . Author mentions being motivated by 3 theorists, two of whom are mentioned in wikipedia article Barad and Pickering.
  - in software engineering, people often talk about a program representing knowledge about a domain from the point of view of how the designers and maintainers work with that knowledge. in this article, we are asked to consider the application as a source of knowledge for the user.
  - a 20 minute video by the first author of related interest is: Interactive Performance as a Means of Civic Dialogue https://www.youtube.com/watch?v=VJwM_Ppgf7c (CHI 2017 talk so it comes after this paper) a mobile app was used to mediate the performance. this was a published paper at SIGCHI 2017, see publication list for Normark at https://dl.acm.org/author_page.cfm?id=81100425412&coll=DL&dl=ACM&trk=0 (2 Ecofriends papers)
  - see publication list for Tholander (second author) https://dl.acm.org/author_page.cfm?id=81100187595&coll=DL&dl=ACM&trk=0 (3 Ecofriends papers)

---

**Marking rubric for presentations**

*based on Evaluating oral presentations (paper)*

---

**Presentation Evaluation -- CS9622 -- 4 Dec 2017**

**Name as appears on Student ID:**

**UWO computer username:**

**Title of Talk:**

**Hand back to prof without filling in below!**

==========================================================================

1) Effective use of slides:

2) Voice dynamics, eye contact:

3) Appear comfortable:

4) Presentation appears well prepared:

5) Introduces self, overviews talk:
6) Slides focus on main point, complements talk rather than distracts:

7) Interestingness of topic clear:

8) Review, future directions, full references:

9) Evaluation of overall presentation:

   total (of 90 -- each out of 10):

   time taken (10 - 15 minute talk target):

---

**Ninth class (Monday) 13 Nov 2017**

**Music for the Ninth:**
- Joseph Szigeti - Die Biene (The Bee) op. 13, Nr. 9 - Franz Schubert [https://www.youtube.com/watch?v=ST-BeH4Ak5vg](https://www.youtube.com/watch?v=ST-BeH4Ak5vg)
- François Schubert: Die Biene (The Bee) (Jan Van Hoecke and Pieter Dhooore) [https://www.youtube.com/watch?v=1h06i4jhdI](https://www.youtube.com/watch?v=1h06i4jhdI)
- Lang Lang —— Flight of the Bumblebee [https://www.youtube.com/watch?v=ptRsnhiP1dc](https://www.youtube.com/watch?v=ptRsnhiP1dc)
- Malena Ernman & Martin Fröst in Flight of the Bumblebee [https://www.youtube.com/watch?v=sbQwQetKm2g](https://www.youtube.com/watch?v=sbQwQetKm2g)
- Flight of the Bumblebee - The Swingle Singers [https://www.youtube.com/watch?v=CZo1_BDn8Ew](https://www.youtube.com/watch?v=CZo1_BDn8Ew)
- The King's Singers - Flight Of The Bumblebee [https://www.youtube.com/watch?v=hFyRruZLchM](https://www.youtube.com/watch?v=hFyRruZLchM)
- Liberece Flight of the Bumblebee and The Bumble Boogie [https://www.youtube.com/watch?v=04u5sioNek8](https://www.youtube.com/watch?v=04u5sioNek8)
- ROBOCLARINET [https://www.youtube.com/watch?v=mAIeTm4lO5Q](https://www.youtube.com/watch?v=mAIeTm4lO5Q)

Followup on music from end of last class:
- https://www.youtube.com/watch?v=JTEFKFkISx4 John Cage's 4'33"
- https://www.youtube.com/watch?v=Oh-o3udlm8 4'33" John Cage(Ochestra with Soloist, K2Orch, Live) / 433
- https://www.youtube.com/watch?v=WTCvKfKnKl5 Is John Cage's 4'33" music?: Prof. Julian Dodd at TEDxUniversityOfManchester (useful background even if conclusion wrong)

**reminders:**
- still sorting out timing of final exam (see Eighth class (Monday) 6 Nov 2017)
- last progress report due today
- next week, 3 minute presentations
- Toward the final exam

**an interesting 3 minute video:** Conducting a Systematic Literature Review [https://www.youtube.com/watch?v=WUEribfXV0](https://www.youtube.com/watch?v=WUEribfXV0)

**a related notes is:** What is a “mapping study?”. Diane Cooper, J Med Libr Assoc. 2016 Jan; 104(1): 76–78 [https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4722648/](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4722648/)

**four minutes on sustainability:** W Daniel Hills – The Clock of the Long Now (176/248) [https://www.youtube.com/watch?v=g-5ZozVnHfw](https://www.youtube.com/watch?v=g-5ZozVnHfw) (see also: Danny Hills: The Internet could crash. We need a Plan B (12 min) [https://www.youtube.com/watch?v=AOEQ9GteWbg](https://www.youtube.com/watch?v=AOEQ9GteWbg))

**a related note is:** BBC Future Technology Japan earthquake spurs technology innovation (6:06) [https://www.youtube.com/watch?v=7rWeLHPZQUs](https://www.youtube.com/watch?v=7rWeLHPZQUs)

**readings:**
- One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques (paper)
- Towards Enhanced Usability of IT Security Mechanisms - How to Design Usable IT Security Mechanisms Using the Example of Email Encryption (paper)
- Exploring language technologies to provide support to WCAG 2.0 and E2R guidelines (paper)
  - [https://www.w3.org/WAI/WD/easy-to-read](https://www.w3.org/WAI/WD/easy-to-read)
  - [https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning.html](https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning.html) (readability)
  - [https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-idioms.html](https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-idioms.html) (unusual words)
and a set of three papers, three to four pages long each from the same research group on what allows software to sustain:

- Toward Architecture Knowledge Sustainability: Extending System Longevity (paper)
- Relating Architectural Decay and Sustainability of Software Systems (paper)
- Identifying Architectural Bad Smells (paper)

and of course notes re readings for 13 Nov 2017

notes re readings for 13 Nov 2017

- One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques (paper)
  - see second column first page "The goal of this research is ..."
    - http://cwe.mitre.org/about/index.html
    - http://cwe.mitre.org/
- Towards Enhanced Usability of IT Security Mechanisms - How to Design Usable IT Security Mechanisms Using the Example of Email Encryption (paper)
  - security is often an afterthought and usability of that security is an afterthought of an afterthought that everything depends on
  - 9 guidelines for usable security design
      - http://www.usabilitynet.org/tools/r_international.htm
      - https://www.userfocus.co.uk/resources/iso9241/intro.html
      - http://www.usabilitynet.org/tools/r_international.htm
      - https://www.w3.org/2002/Talks/0104-usabilityprocess/slide3-0.html
      - http://www.testingstandards.co.uk/usability_guidelines.htm
  - email encryption example
  - forced updates
  - captchas – accessibility problems
  - browser certificate validation
- Exploring language technologies to provide support to WCAG 2.0 and E2R guidelines (paper)
  - readability is a accessibility requirement in WCAG 2.0
    - cf third paragraph above Section 3.1 quote: "important sentences appear in specific places, for instance in introduction and conclusion sections of a paper"
  - https://en.wikipedia.org/wiki/Gunning_fog_index
  - https://en.wikipedia.org/wiki/Flesch%E2%80%93Kincaid_readability_tests
  - https://www.w3.org/WAI/RD/wiki/Easy_to_Read
  - https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning.html (readability)
    - https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-idioms.html (unusual words)
    - https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-located.html (abbreviationls)
    - https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-supplements.html (reading level)
    - https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-pronunciation.html (pronunciation)
- also of interest
  - https://grammark.org/dist/#/
  - https://languagetool.org/
  - https://www.grammarly.com/
  - https://alternativeto.net/software/grammarly-grammar-checker/?platform=linux
- and a set of three papers, three to four pages long each from the same research group on what allows software to sustain:
  - Toward Architecture Knowledge Sustainability: Extending System Longevity (paper)
    - references tracking architecture knowledge in `decision networks' and `decision models'
    - defines `architectural sustainability'
    - define architectural smells
    - ripple effect metrics
    - define architectural instability
    - Figure 1 smells derived from metrics
    - Table 1 metrics – approach is to customize a combination of metrics
  - Relating Architectural Decay and Sustainability of Software Systems (paper)
    - Table II architectural smells and metrics
    - see conclusions
  - Identifying Architectural Bad Smells (paper)
defines architecture as collection of important design decisions
https://en.wikipedia.org/wiki/List_of_software_architecture_styles_and_patterns
https://en.wikibooks.org/wiki/Introduction_to_Software ENGINEERING/Architecture/Anti- Patterns
two example architectural smells

Second Class (Monday) 18 Sept 2017

( Where Dreams are Made (A.I. Artificial Intelligence) https://www.youtube.com/watch?v=Y12wjPuUhP1 )

Sigh. Forget to mention the links for Today's material that were at the bottom of First Class (Monday) 11 Sept 2017, i.e.,

- on systems: In A World of Systems (video) (9 minute video)
- on requirements: An introduction to Requirements Engineering (talk) (10 minute talk)
- on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
- on accessibility: Universal Design (talk) (24 minute talk)
- on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)

These links were/are meant to give a bit more depth on various topics of interest to use this semester. The intent is that in the future there will always be links like this letting people know what is coming as a whole week passes between each meeting.

Also, from last class, we have the unresolved issue:

- NOTE: Final Exam date was not in the course outline. The response was that it would probably be 4 Dec if there was room for it and the presentations that day and, if not, it would be the week before.

which also now appears as a comment on the course outline (see bottom of page Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability)

Adminstratrivia: MC 300 is usually locked. If you arrive for class and I haven't yet unlocked it yet, you can check for me at my office MC384.

Followup on Dvorak keyboard: Why the Dvorak keyboard didn't take over the world (talk) and OS X Yosemite: Use Dvorak keyboard layouts https://support.apple.com/kb/PH18382?locale=en_US

Followup on chord keyboard and mouse: the first mouse (demo)

Followup on designing cities for both young and old: 8 80 Cities (Toronto web site)

Searching the Computer Science Literature

Since a project proposal is due 2 Oct 2017 (see Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability), it is worth looking at Thesis Projects A Guide for Students in Computer Science and Information Systems (book) which contains a chapter Developing Your Project Proposal https://link.springer.com/chapter/10.1007/978-1-84800-009-4_5. While the proposal you will need for this course is smaller than you would be expected to do for a thesis, it follows basically the same form as both proposal types address typical issues arising in undertaking a project.

Getting back to the material referred to at the top of this note:

- on systems: In A World of Systems (video) (9 minute video)
- on requirements: An introduction to Requirements Engineering (talk) (10 minute talk)
- on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
- on accessibility: Universal Design (talk) (24 minute talk)
- on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)

as a grad student, you are expected to be able to read material like this and figure out what it is about. Four questions that you should be considering as you look at this material are:

- What is the main point the author/speaker is trying to get across?
- Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into what else you know about this topic?
- Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new vocabulary help organize the presentation?
- Can you see how this information might be used by you in some future undertaking?

If you are reading this before class, try and answer these questions for yourself before looking at my Notes on readings for second class

Where are we headed next? Get feedback from class.

The results can be viewed as a Multistakeholder Requirements for Next Class
software sustainability wrt economics and technology
sustainable development of HCI
sustainability for critical systems -- failure handling
integration of safety and security requirements; sustainability in software engineering; sustainability for big data
maintainability as an aspect of sustainability
safety and network security
safety issues in software development
safety/security with sustainability
software security; big data/cloud security
security and software engineering
accessibility of e-commerce web sites; accessibility for children and elders
how can this course help regarding programming?
  • gives a wider view of how programming fits into the world
  • observes that programming problems often have their roots in requirements gathering and introduce approaches that seek to
    minimize these problems
  • observes how programs fit into society and the impacts that result from too narrow an understanding of what needs to be
    programmed and introduce approaches that seek to minimize these problems
  • introduce tools specific to programming in areas where various non-functional requirements are recognized as important. such
    tools become particularly important in environments that use automated testing frameworks.

notes:

  • note: names are friendly
  • note: email addresses are useful as they allow followup and are unique identifiers
  • note: student id numbers are a nuisance due to associated privacy/security problems

• Rather than put the readings/videos here, I am putting them in the material for the next class Third Class (Monday) 25 Sept 2017

Notes on readings for second class

Getting back to the material referred to in Second Class (Monday) 18 Sept 2017

  • on systems: In A World of Systems (video) (9 minute video)
  • on requirements: An introduction to Requirements Engineering (talk) (10 minute talk)
  • on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
  • on accessibility: blog entry on recent US court ruling http://www.webaxe.org/landmark-web-accessibility-ruling-in-u-s/
  • on accessibility: Universal Design (talk) (24 minute talk)
  • on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)

as a grad student, you are expected to be able to read material like this and figure out what it is about. Four questions that you should be
considering as you look at this material are:

  • What is the main point the author/speaker is trying to get across?
  • Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into
    what else you know about this topic?
  • Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new
    vocabulary help organize the presentation?
  • Can you see how this information might be used by you in some future undertaking?

If you are reading this before class, try and answer these questions for yourself before looking at my Notes on readings for second class 170906
170906

😊

see Notes on readings for second class

  • What is the main point the author/speaker is trying to get across?
  • Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into
    what else you know about this topic?
  • Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new
    vocabulary help organize the presentation?
  • Can you see how this information might be used by you in some future undertaking?

regarding:

  • on systems: In A World of Systems (video) (9 minute video)
    • shows you how to see the world as a collection of systems that interlock and are part of larger systems
defines systems in terms of parts, connections, and underlying goal

• gives examples of feedback loops
• defines balancing feedback loop
• observe importance of delays in the loops and notation for indicating such
• introduced terms stock, in flow, out flow
• introduces notion of reinforcing feedback loop

• on requirements: An introduction to Requirements Engineering (talk) (10 minute talk)
  • tells us why requirements engineering is important - see particularly 5:38
  • wants to view things in terms of user world, requirements, and software-based system
  • does system mean the same thing here as it did in the previous talk? how do loops figure in?
  • defines requirements
  • introduces functional, nonfunctional, and domain requirement types
  • defines requirements engineering – systematic repeatable techniques for producing good requirements documents
  • lists causes of difficulties in getting requirements right

• on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
  • recall the usage of constraints in the previous talk and see how it is used in this paper.
  • note how loops in a system figure into this paper

• on accessibility: blog entry on recent US court ruling http://www.webaxe.org/landmark-web-accessibility-ruling-in-u-s/
  • what does it mean that a website is a place of public accommodation? fortunately there is google https://www.google.ca/search?q=%22a+place+of+public+accommodation%22&oq=%22a+place+of+public+accommodation%22 . Note different people get different results from same search. My results included
    • http://www.3playmedia.com/2016/06/08/earl-v-ebay-is-a-website-a-place-of-public-accommodation/ which discusses earlier rulings involving eBay and Netflix that had gone the other way.
    • https://www.eeoc.gov/facts/adafaq2.html defining the term for the relevant law
    • https://www.ada.gov/ada_title_III.htm the relevant law
    • https://www.lexology.com/library/detail.aspx?g=149404c6-d384-477a-84e3-8067ebc584b a different case that agreed online is a public accommodation
  • $250,000 to make website accessible. $2 million spent to build it in 2016 and $7 million spent to upgrade it in 2017.

• on accessibility: Universal Design (talk) (24 minute talk)
  • talk by Valerie Fletcher, Institute for Human Centered Design http://www.humancentereddesign.org/
  • 2:00 quote: 2 core ideas
    • Design powerfully and profoundly influences everyone and our sense of confidence, comfort, and control
    • Variation in ability is ordinary, not special, and affects most of us for at least part of our lives.
  • defines effective communication – strengths of work so far on visual but weak on audio
  • connects universal design to social sustainability https://en.wikipedia.org/wiki/Social_sustainability
  • US federal government uses over 60 different definitions of disability.
  • universal design, inclusive design (UK), and design-for-all all refer to the same general idea
  • 11:04 definitional quote: a framework for the design of places, things, information, communication and policy that focuses on the user, on the widest range of people operating in the widest range of situations without special or separate design
  • it is the environment (part of which we design) that makes a disability disabling
  • example of bad design of medicine bottles and possible alternatives (15:15)
  • http://www.gallaudet.edu/campus-design-and-planning/deafspace
  • Design Meets Disability (talk) (book)

• on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)
  • defines sustainability as "the capacity to endure"
  • defines sustainable development as "meeting the needs of the present without compromising the ability of future generations to meet their own needs"
  • ACM author page: http://dl.acm.org/author_page.cfm?id=81384610283
  • scholar google page: https://scholar.google.ca/citations?user=TNoA6ZMAAAAJ
  • author home page: http://birgit.penzenstadler.de/
  • https://www.iso.org/iso-14001-environmental-management.html
  • Systematic mapping study on software engineering for sustainability (SE4S) (paper)
  • section 3 reference to "principles of systems thinking"

Where are we headed next? Get feedback from class. Results at: Second Class (Monday) 18 Sept 2017

Seventh class (Monday) 30 Oct 2017

Music for a Monday:

• James Bond 007 Theme Tune (original) 1:44 https://www.youtube.com/watch?v=ye8KyYK9n-0
• BBC Proms 2011: James Bond Theme 1:54 https://www.youtube.com/watch?v=J9-cDa4JcWM
• Robot Quadrotors Perform James Bond Theme 1:39 https://www.youtube.com/watch?v=_sUeGC-8dyk
Followup: The unruly power grid (paper) and Notes on Ontario and 2003 North American Power Failure

Topics requested:

- Web accessibility
- Difference between usability and accessibility
- Nonfunctional requirements for Internet of Things
- Cryptography and cloud data security
- Security testing in Java or Javascript
- Sustainable software and sustainable software engineering process
- Sustainability in critical systems
- Design rules/methodologies for minimizing impact of fatal failures in critical systems

Readings:

  Section 6.3 pages 151 – 154, and Chapter 7 up to and including 7.4.1 (so up to page 170 where Section 7.4.2 starts)
- A survey of compliance issues in cloud computing (paper)
- GreenOracle: estimating software energy consumption with energy measurement corpora (paper)
- Building a safety architecture pattern system (paper) – note we are only interested in the main body of the paper (just over 10 pages) and not the 30+ pages of appendix.

with the usual comments Notes for seventh class readings

Notes for seventh class readings

- Engineering a Safer World, Nancy Leveson, pp 151 -- 170
  - 6.3 A SYSTEM SAFETY ENGINEERING PROCESS
  - safety is about enforcing constraints on the behavior of components of the system
  - Figure 6.1 The Components of a System Safety Engineering Process Based on STAMP p; 152
  - 6.3.1 Management
  - management is central in shaping behavior of humans envolved in system
  - 6.3.2 Engineering Development
  - safety must be involved from the beginning of concept formation stage of development
  - see Chapt 7 for details on how to identify safety requirements and constraints
  - Chap 8 describes new hazard analysis technique
  - Chap 9 describes general principles for safe design
  - Chap 10? -- note disconnect between sum up sentence at paragraph end and what went before
  - operators also need to be in early design process as they are familiar with how they use these sorts of systems
  - 6.3.3 Operations
  - a process for managing safe operations is discussed in Chapt 12
  - Chapt 7 FUNDAMENTALS
  - 7.1 Defining Accidents and Unacceptable Losses
  - include Rationale: and Assumption: where appropriate
  - assign levels of serverity where appropriate
  - customers may have general safety policies for contractors, see NASA example pp 157
  - 7.2 System Hazards
  - hazard: a system state that will result in an accident
  - failures is reliability term whereas hazard is a safety term
  - Hazard + Environmental State => Accident (Loss)
  - incident: hazard without loss (benign environment)
  - 7.2.1 Drawing the System Boundaries
  - system boundary is limit of control by system designer
  - top down design is necessary to figure out which aspects of system safety are to be controlled by which components
  - 7.2.2 Identifying the High-Level System Hazards
  - usually there are less than a dozen
  - then map system level hazards to component level hazards
  - more on refinement process in Chapt 9
  - domain expertise and subjective evaluation required
  - Chapt 13 presents common heuristics used
  - what matters is what the system stakeholders agree to
  - 7.3 System Safety Requirements and Constraints
  - example Fig 7.1 Design Constraints for train door hazards pp. 163
  - requirements and constraints are then created to avoid hazards
  - resolving conflicts between constraints is important aspect of design process
  - pp 164 Fig 7.2 High-level requirements and design constraints for air traffic control
  - pp. 165 Fig 7.3 High level design constraints for TCAS
  - related constraints in different components raise coordination issues
  - 7.4 The Safety Control Structure
• additional safety requirements and constraints address the organizational and social system issues of operations and maintenance
• some general principles for this are addressed in Chapt 13
• 7.4.1 The Safety Control Structure for a Technical System
• NASA safety revisions after Columbia Incident
• A survey of compliance issues in cloud computing (paper)
  • sidenote: ideally requirements come from negotiations among stakeholders but when not practical, standard 'requirements' can be set by the government on behalf of stakeholders (cf hearings on proposed regulations where stakeholders can get some voice in process)
  • regulations “are intended for lawyers, not for software developers”
  • from abstract: advocating the need for reference architectures for regulations
  • what would such a thing look like? the word reference appears 62 times
  • they should be vendor neutral standards
  • will use abbreviation RA
  • [9, 34, 35, 49, 51, 61] give examples
  • the paper wants to discuss what should be included in a regulation compliance RA
  • current state of art is internal IT auditors and independent third party certifying agencies (third party assessment organization 3PAO)
  • grammar note: Reference [45] is bad writing. [45] is a way of representing a superscript and is not a 'word' in the sentence, but markup on top of the sentence. The usual phrasing is to refer to the authors of entry 45 in the bibliography, like: Mirkovic [45] observes ...
• 2.3 Reference Architectures (RAs)
• 3 Survey pp 4 -- pp 8
• 4 Compliance approaches in industry
• service providers have compliance architectures specific to their proprietary cloud platforms
• 5 Completeness of compliance issues and recommendations
• 5.1 Dictionary of regulations
• 5.2 Regulation overlaps
• 5.3 Lack of standard Reference Architectures (RAs)
• 5.4 Lack of control and transparency
• 5.5 Security threats
• [22] security reference architecture for cloud security
• 5.6 Overlap with security
• regulations often about security issues
• need unified approach to security and compliance
• writing notes:
  • 1) see how copied information from papers contains citation, e.g., Table 5. for publication, authors sometimes need to be contacted to get permissions and some journals give a standard permission requiring specific citation practices.
  • 2) even though Section 3 is a bunch of paper summaries with some comparisons, the paper as a whole is driven by the theme of trying to justify the need for work on common reference architectures for regulations -- the value added.
• GreenOracle: estimating software energy consumption with energy measurement corpora (paper)
  • measuring energy consumption is difficult for developers.
  • collect a lot of energy measurements to create a model that can be used to estimate energy consumption within 10%
  • model is not application specific
  • system calls and cpu utilization end up being key features needed for estimates
  • (good abstract)
  • pp 57 Section 6: Developer's Workflow to Estimate and Improve Energy Consumption
• Building a safety architecture pattern system (paper) – note we are only interested in the main body of the paper (just over 10 pages) and not the 30+ pages of appendix.
  • Fig 1 Explanation of the basic GSN elements
  • Table III Determining the tactics used by the TMR patterns ...
  • one approach to software faults is multiple independent implementations
  • N-Version Programming Pattern pp 22
  • Acceptance Voting Pattern
  • Recovery Block Pattern
  • another approach is to shut down when problems detected
  • N-Self Checking Programming Pattern
  • Sanity Check Pattern
  • Monitor Actuator Pattern
  • Watchdog Pattern (specific to timing issues)
  • or you can have a 'fail-safe state' to go to rather than a full shutdown
  • Safety Executive Pattern
  • Protected Single Channel
  • or a fail-safe with shut down backup
  • 3-Level Safety Monitoring
  • pp 37 Appendix B Safety Tactics
  • keep it simple
  • condition monitoring
  • comparison
  • repair
  • override
  • barrier
  • regarding fail-safe see references (known uses and credits) for patterns where they are referred to: Sanity Check Pattern, Monitor-Actuator Pattern, Watchdog Pattern, Safety Executive Pattern, Protected Single Channel, and 3-Level Safety Monitoring
Sixth Class (Monday) 23 Oct 2017

music:

- Computer Controlled Orchestra https://www.youtube.com/watch?v=xAOX8W3D6Mg
- Atom-powered Intel Industrial Controller in Concert playing Pipe Dream  https://www.youtube.com/watch?v=E4hjx3_A-cw
- Intel Computer Controlled Orchestra - Maker Faire Bay Area 2012 https://www.youtube.com/watch?v=57kEK6eKkyM
- The Robotic Musicians known as: Intel's Industrial Control in Concert | Intel https://www.youtube.com/watch?v=JLdB0WEixjM

followup on music from fifth class: 100 Year Old Self-Playing Violin - "The Eighth Wonder Of the World"  https://www.youtube.com/watch?v=xS0mP2cOmJs shows where the idea for the marble machine came from

On writing literature reviews

- Although not Computer Science specific, this paper addresses many important high level issues in thinking about putting together a literature review How to Write a Literature Review Paper? (paper)
- Section 2: The Rationale for Writing an LRP and Added Value
  - quote A sign of a good review is that the value added permeates the whole review and not just the conclusions.
  - note the authors are using the pseudo-economic term 'value', but one can also see this in terms of documents being written for a purpose. An LRP exists in a system including the author and the reader and the document itself. Generally the author writes to make people aware of things the author is interested in and in increase the general social interest (and support) for those things. Generally, the reader reads because there is a problem the reader wants to solve and they think the document will help them in this task. The 'added values' can be read as the kinds of information that helps people with various tasks. In this article, the reader is viewed as being part of the research community in the area that the document discusses. For our writings, it would be unusual for someone to intend to publish it. The tasks that the document should, ideally, be aiding are thus tasks the writer expects to be undertaking. Figuring out the intent (proposed added value) of the document gives focus to the structure of the document as well as guides decisions such as what literature is relevant.
- Section 3: Types of LRPs
- Section 4: Methodology: Selection of Papers
  - see paragraph on snowballing (forward and backward)
- Section 5: Structure of paper

Readings:

- Requirements and testing: seven missing-link myths (paper)
- Sustainable E2 mobility services for elderly people — Platform system architecture (paper) (recommender system)
- Erlang (paper) – failure handling in systems that are supposed to run without stopping, such as telephone switching network
- One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation (paper) – cloud (virtualization) security and underlying hardware

with the usual comments, see Notes re readings for sixth class

Notes re readings for sixth class

  - once more we review systems, organized complexity, open systems, emergence and hierarchy, and communication and control. connect it to safety distinct from reliability. the importance of constraints. applied to understanding accidents, assumptions of systems engineering, and designing for safety. from here we will skip forward to 6.3 to see how STAMP, STPA, and CAST actually work.
  - worthwhile looking at how it is written.
    - in general, sentences are too long – magic number 7 +/- 2 https://en.wikipedia.org/wiki/The_Magical_Number_Seven,_Plus_or_Minus_Two
    - notice strong use of connective passages linking text together in a logical structure How to Create a Concept Map https://www.youtube.com/watch?v=sZJj6DwCqSU 2:00
    - notice heavy use of examples making the discussion concrete so reader doesn't get lost in abstractions
- Requirements and testing: seven missing-link myths (paper)
- connection to TDD Test Driven Development https://en.wikipedia.org/wiki/Test-driven_development
- myth 3 and the importance of concrete examples in communication among stakeholders as well as testing
- Robertson's fit criteria
  - https://needsmust.wordpress.com/2011/05/04/fit-for-purpose/
• myth 6. a tester's job, the oracle assumption
• parallel between requirements for time points in time plan and the question of how to test to see if these have been accomplished.
• Sustainable E2 mobility services for elderly people — Platform system architecture (paper) (recommender system)
• difference between accessibility and other uses of the word access https://scholar.google.ca/scholar?q=accessibility+audit+of+recommender+systems&as_sdt=0%2C5&q=accessibility+audit+of+recommender+systems&as_vis=0&hl=en&as_sdt=0%2C5&as_q=accessibility+audit+of+recommender+systems&as_vis=0&hl=en&as_sdt=0%2C5
• https://scholar.google.ca/citations?user=3JiQmlcAAAAJ&hl=en&oi=sra (accessibility and educational support software)
• how could success of this system be evaluated given what is said in Results section? does Fit Criteria help?
• Erlang (paper) — failure handling in systems that are supposed to run without stopping, such as telephone switching network
  • https://en.wikipedia.org/wiki/Erlang_(programming_language)
  • immutable store
  • pp 69: Key Insights
  • pure protocols
  • fault tolerance – compare to safety criteria
  • pp. 70: observing process detects errors
  • four key properties
  • multicore
  • Erlang world view
  • Erlang view of errors
  • layered system, compare hierarchy and control
  • pp. 72 detecting errors
  • Let It Crash
  • pp. 73 dynamic code upgrade (sustainability)

• OTP Open Telecoms Platform https://en.wikipedia.org/wiki/Open_Telecom_Platform

• One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation (paper) – cloud (virtualization) security and underlying hardware
  • 2016 paper
  • when we require something to be secure, what do we really want?

**Tenth class (Monday) 20 Nov 2017**

**reminder:**

• we are doing 3 minute practice presentations today.

**music:**

• Mussorgsky’s “Night on Bald Mountain” - Ludwig Symphony Orchestra  https://www.youtube.com/watch?v=XyR-poMsSWI
• Night On Bald Mountain - Fantasia (1941) (Theatrical Cut) https://www.youtube.com/watch?v=SLCuL-K39eQ
• Alexeiiff's Night on Bald Mountain (great sound) https://www.youtube.com/watch?v=wYbjW7XrWDo
  • Here and the Great Elsewhere - Michèle Lemieux and the Secret of Pinscreen (Making of) https://www.youtube.com/watch?v=N_pDZY8AuaQ
• Wolf Hoffmann (ACCEPT) Night On Bald Mountain Official Music Video

**note on bibliographies:**

• in general, bibliography looks like taken from other articles that used varying conventions. bibliography should be consistent. don't capitalize doi in one reference and not in another and such. provide author, title, journal/conf where available, organization that published it where not in journal/conf. page numbers or page count are useful as well.

• How Bibliographies, Citations, etc. are Handled in Computer Science (note)

• bibliographies are difficult if you haven’t been trained to do it. that is why people use proper typesetting systems like TeX and LaTeX that come with tools such as bibtex
  • https://en.wikipedia.org/wiki/BibTeX
  • http://www.bibtex.org/
  • https://www.reed.edu/cis/help/LaTeX/bibtexstyles.html

what are you using to manage your bibliographies?

**class material:**

• Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems (paper)
• Why do some (weird) people inject faults? (paper)
• Prejudices, memories, expectations and confidence influence experienced accessibility on the Web (paper)
• Quality Requirements in Agile as a Knowledge Management Problem: More than Just-in-Time (paper)

with the usual comments: Notes on the readings for Tenth class (Monday) 20 Nov 2017

**Notes on the readings for Tenth class (Monday) 20 Nov 2017**
Third Class (Monday) 25 Sept 2017

(Blade Runner Blues - Vangelis - 1982 - [HD] https://www.youtube.com/watch?v=RScZrvTebeA)

Carrying forward from the end of last class:

Where are we headed next? Get feedback from class.

The results can be viewed as a Multistakeholder Requirements for Next Class

- software sustainability wrt economics and technology
- sustainable development of HCI
- sustainability for critical systems – failure handling
- integration of safety and security requirements; sustainability in software engineering; sustainability for big data
- maintainability as an aspect of sustainability
- safety and network security
- safety issues in software development
- safety/security with sustainability
- software security; big data/cloud security
- security and software engineering
- accessibility of e-commerce websites; accessibility for children and elders
- how can this course help regarding programming?
  - gives a wider view of how programming fits into the world
  - observes that programming problems often have their roots in requirements gathering and introduce approaches that seek to minimize these problems
  - observes how programs fit into society and the impacts that result from too narrow an understanding of what needs to be programmed and introduce approaches that seek to minimize these problems
  - introduce tools specific to programming in areas where various non-functional requirements are recognized as important. such tools become particularly important in environments that use automated testing frameworks.

notes:

- note: names are friendly
- note: email addresses are useful as they allow followup and are unique identifiers
- note: student id numbers are a nuisance due to associated privacy/security problems

Looking at these results, we see security coming up a lot as well as various aspects of safety and sustainability. If we had to choose just one topic going forward, the situation would be complicated. Indeed, there is a whole area of study regarding how to combine the preferences of individuals into some notion of ‘what does the group want’, see: https://en.wikipedia.org/wiki/Social_choice_theory and https://en.wikipedia.org/wiki/Arrow%27s_impossibility_theorem. However, I am going to use this to choose a few different topics for the third and fourth class meeting and then raise the question again to see where people think we should go next and how well I am interpreting these results.

READINGS FOR THIRD CLASS:

- Let’s move a bit deeper into understanding Leveson’s view on safety by looking at Software Challenges In Achieving Space Safety (paper)
- An interesting way to see system level issues manifesting in security is through the concept of covert channels https://en.wikipedia.org/wiki/Covert_channel and in particular timing attacks https://en.wikipedia.org/wiki/Timing_attack. A nice example of this is Timing Analysis of Keystrokes and Timing Attacks on SSH (paper) from 2001.
- A look at how sustainability and maintainability interact can be found in Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper)
As noted in the previous class as a grad student, you are expected to be able to read material like this and figure out what it is about. Four questions that you should be considering as you look at this material are:

- What is the main point the author/speaker is trying to get across?
- Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into what else you know about this topic?
- Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new vocabulary help organize the presentation?
- Can you see how this information might be used by you in some future undertaking?

Let's look at a few of these questions in a bit more detail to see what the fastest way to answer them might be:

- What is the main point the author/speaker is trying to get across?
  - Where do you find the statement of the main point of the author? We often see presentation structures like Abstract, Introduction, Main Body, Conclusion. Frequently Abstract is just a summary of Introduction. Often Introduction is about two things: 1) motivating the reader to continue on and 2) provide some background definitions. If the main point appears in the Abstract or Introduction, it is often vague and easy to misinterpret (people sometimes exaggerate their claim in the Intro to get people to read further – something you should avoid). The Main Body is large and occasionally doesn't even contain the main point (being primarily focussed on the evidence for the main point). Most of the time I find the clearest statement of the main point in the Conclusion section. So usually I start my reading at the conclusions. That way I quickly know what the paper is all about and can start to think about what information I want to gather from the rest of the paper.
  - Things don't always work out the way you expect. Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper) doesn't have a conclusions section. However, it does have a section called: Research Hypothesis and Goal which is just what we are looking for.

- Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into what else you know about this topic?
  - The evidence for the main point often clarifies what the author/speaker means by the main point. Sometimes it is clear and the Did you already agree and How does this information fit questions can be answered just from figuring out what the main point is.
  - However, most of the time the evidence is the interesting part of the paper. So you can skim through the paper looking for things like tables of experimental results (or figures). A person who wants to use the work or extend the work needs to determine how to reproduce the evidence done which often requires tracking down references etc. But before one gets to that there is the notion of just identifying the structure of the way the author/speaker is arguing for their main point and asking if you can think of possible gaps or alternative interpretations of the evidence which would cause you to discard the main result even if the evidence presented was correct.
  - Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper) is a good paper to consider alternative ways you can interpret the evidence for the main claim.

- Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new vocabulary help organize the presentation?
  - Usually new vocabulary appears up front in the Abstract/Introduction part of the paper. If you are lucky, it will appear in bold font or italic font or in quotation marks or in phrases like `We define ...` or `... means ...``
  - Since these papers are all online, you can use search tools to look for things in pdf and/or html versions of the papers to speed up searches. Once you have identified vocabulary of interest, it is worth searching for where in the paper it occurs.
  - In Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper) neither of these tricks worked. But in the title we see two powerful terms that are worth search for references of in the paper: God Class and Power Consumption. God Class appears 20 times in the paper. Power consumption appears 35 times in the paper. Just reading the sentences that include those two words (less than 55 sentences) gives and interesting summary of the paper as a whole.

Reading is an iterative process like software development. In software, you need to sort out every detail to be successful. In reading a paper, it is more about trying to figure out what is going on and if it is of interest to you or not (given your current goals of the moment). For this class, you want to know what people believe about the topics we are looking at, how they justify those beliefs, and what vocabulary they use to discuss those beliefs. In your 'in depth' work you will sometimes want to go deeper into what a paper is saying, but for preparing for class, these three issues are 'far enough in'.

If you are reading this before class, try and answer these questions for yourself before looking at Notes on Readings for Third Class (25 Sept 2017)

Proposal Handin Notes:

from the course outline:

- 2 Oct 2017: Proposal for In Depth work due
- Both approaches can be seen as involving four stages: 1) a proposal of what is to be undertaken (5%); 2) the undertaking (with progress reports (3 progress reports, 5% each)); 3) the writeup of the undertaking (50%); and 4) the presentation/demo of the undertaking (10%).

from chapter on developing your project proposal: https://link.springer.com/chapter/10.1007/978-1-84800-009-4_5

- the proposal should be short and easy to read (in particular, easy for the marker to find the stuff they are looking for). use one of the two structures for section headings shown in the chapter
  - pp.27 Subject area, Aim, Arguments, Objectives
  - pp.34 Introduction, Reasons, Aim of Project, Objectives
  - for either structure, for the proposal as a whole, you should give it a meaningful title and include your name and email address -- no need for student ID numbers
The claims that structure the argument in support of the thesis statement:

- Specific goals for each progress report are needed in order to be able to monitor progress and know when things need to be rethought.
- Resources are important. What background do you have in the area chosen?
- Resources are important. You are probably looking at a 60 hour project. Is it realistic to undertake a measurable amount of progress on your aim in that time frame?
- Resources are important. You are probably looking at a 10 page write-up at the end. Is your project structure well enough organized that it can be reasonably explained in that amount of space?
- Resources are important. Non-programming projects generally involve reading papers. How long does it take you to read a paper (I generally budget 5 minutes a page for myself, but you may read faster or slower than that – to make plans for your own work, you should know how long things take you).
- Resources are important. Programming projects are notoriously difficult to estimate the amount of time it will take to do things. A programming project should have an iterative structure, so that if you have to stop early there will still be things done that can be written about and illustrate issues related to the course.
- Resources are important. Writing, like programming, is notoriously difficult to estimate the amount of time it will take to do things well. Budgeting write-up time for the end of the project is a recipe for disaster. Projects get judged on write-ups and presentations. So good project work and poor write-ups become poor marks. The solution is to ALWAYS BE WRITING (cf Always Be Closing 0:12 https://www.youtube.com/watch?v=Yz246_Pjjkc). As you complete sub goals, include write-ups of those sub goals to eventually become the first draft of the final write-up. Also, remember the quote from Leslie Lamport: Thinking Above the Code https://www.youtube.com/watch?v=4YP3j_jk8Q at 3:27, "To think you have to write. If you're thinking without writing, you only think you're thinking." Writing papers in computer science

Notes on Readings for Third Class (25 Sept 2017)

Readings for Third Class:

- Let’s move a bit deeper into understanding Leveson’s view on safety by looking at Software Challenges In Achieving Space Safety (paper)
- An interesting way to see system level issues manifesting in security is through the concept of covert channels https://en.wikipedia.org/wiki/Covert_channel and in particular timing attacks https://en.wikipedia.org/wiki/Timing_attack. A nice example of this is Timing Analysis of Keystrokes and Timing Attacks on SSH (paper) from 2001.
- A look at how sustainability and maintainability interact can be found in Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper)

Software Challenges in Achieving Space Safety -- Nancy G. Leveson

The last sentence contains the thesis statement of what the author is advocating: quote: “System safety techniques and other spacecraft engineering activities need to be expanded to include both software and the complex, cognitive decision-making roles played by human operators.”

What the author means by safety and accidents are the key vocabulary issues in the paper.

Safety is about preventing accidents or losses, either those that damage humans or that damage property or that create excessive costs to correct a deeper understanding of the terms can come from examining the example Fig. 2: The Causal Factors in Spacecraft Losses

The story that keeps on giving: http://www.cbc.ca/news/canada/london/london-ontario-phoenix-pay-western-student-1.4297247 addition to Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018)
1) If we view software as an isolated component, then we must worry about:
   1a) software is more complicated that most people realize and testing is less effective than most people realize
   1b) creating `redundant' software by having isolated groups independently implement from the same specification tends not to improve reliability.

2) But this doesn't really help because usually:
   2a) usually safety problems arise from incomplete software specifications rather than from poor implementation of the specification given.
   2b) software itself isn't dangerous -- it is the control signals the software sends to other components that have the potential to cause harm.

3) The author proposes a new method (Section 2) called STAMP (Systems-Theoretic Accident Model and Processes)
   3a) the safety problem is really a control problem and is caused by inadequate enforcement of safety constraints. these constraints on the interactions among the components of a system define the safety requirements.

4) often the programmer doesn't have enough domain knowlege and the domain expert doesn't realize their hidden assumptions, so incomplete requirements get misinterpreted leading to software that doesn't perform as wanted.

5) similarly, the people worrying about the behavior of the entire system need to know how all the components of the system work and often software is too complicated and gets treated as a simple black box component hiding some of its behavioral assumptions

6) when making critical systems, it is important to keep things simple (KISS). the easy of extension of software leads to `creeping featurism' and excessively complex systems

7) often the system maintainers don't get adequate feedback on problems resulting from software component misbehaving

8) reusing software (often advocated) runs into problems with the need to reanalyze its safety aspects, particularly if the software is proprietary and so not available to the system designers. this has been a cause of many accidents.
   8a) just because software behaved safely in other situations, doesn't help determine if it will be safe in a new situation.
   8b) such reanalysis is expensive increasing the need to keep things simple.

9) sometimes it is thought that computers lessen the knowledge requirements of the human components of a system, but if the humans are controlling what the computer does then they will need more knowledge to include an understanding of the software they are controlling.
10) safety controllers in a system have an internal model of the components they control and when those components are human, there is need to develop a model of human behavior which is generally beyond our current knowledge.

the last sentence contains the thesis statement of what the author is advocating: quote: System safety techniques and other spacecraft engineering activities need to be expanded to

   - include both software 3/4/5/6/7/8 [not software so much as aspects of the software development process, i.e., software engineering]
   - and the complex, cognitive decision-making roles played by human operators 9/10.

How much of this paper's arguments carry through to the earlier paper we looked at by the same author on applying safety analysis to security issues?

Timing Analysis of Keystrokes and Timing Attacks on SSH -- Dawn Xiaodong Song, David Wagner, and Xuqing Tian

This is a 2001 paper. Do you think these problems have been fixed? How?

People type slowly, so ssh ends up sending individual keystrokes out as separate network packets exposing timing data about the user's typing.

That user's typing time patterns are stable is well known and has been proposed as a way of identifying a user as an alternative or second credential to a password.

This is a great example of a paper you don't want to read every word of unless you intend to either reproduce or extend the results presented or use it as the basis for an argument about whether or not ssh is secure.

Analyzing the Harmful Effect of God Class Refactoring on Power Consumption -- Ricardo Pérez-Castillo and Mario Piattini

JDeoderant -- an open source code smell detector for Java programs (reek is an example of an open source code smell detector for Ruby programs)

note how power consumption was measured. pp. 50

at the end, there is a section on limitations of the paper. did they leave stuff out that is relevant? what?
Ruby program for processing exam database
- Script used to build exams from database (runs the ruby program with various parameter settings and invokes pdflatex)
- Example exam database based on material covered in first class (PDF of exam database with each question followed by its answer and PDF of exam database in answer-key format)
- Exam database for questions covering the first 10 classes
- First 10 classes, question and answers pdf
- First 10 classes, questions with answer key pdf
- First 10 classes, 50 randomly chosen questions with answer key
- Questions and Answers for All Eleven Classes pdf
- database of Questions and answers for all eleven classes
- Questions with Answer Key for all eleven classes pdf
- Fifty randomly chosen questions from all eleven classes pdf (much what the final would look like, but I will use different random seeds to chose those questions)

Example exam database based on material covered in first class

```json
{
    "questiontemplate": {
        "question": "What is the answer?",
        "lquestion": "latex reformatting for questions that needed it, absent if not needed",
        "answer": "That which was asked for",
        "lanswer": "latex reformatting for answers that needed it, absent if not needed",
        "note": "optional note",
        "source": "C1"
    },
    "description": "questions from which to construct the Fall 2017 final exam in CS9622a",
    "questions": [
        {
            "question": "ANSWER is the name of the federal payroll system that has regularly appeared in the news since February 2016 due to federal employees not getting paid",
            "answer": "Phoenix Pay",
            "note": "optional note",
            "source": "C1"
        }
    ]
}
```
"question": "CS9622a is primarily about the handling of non-functional requirements with particular focus on requirements associated with safety, accessibility, and ANSWER",
  "answer": "sustainability",
  "note": "optional note",
  "source": "C1"
},
{
  "question": "CS9622 focusses on requirements arising from how software components fit into larger ANSWER",
  "answer": "socio-technical systems",
  "note": "optional note",
  "source": "C1"
},
null
]
Ruby program for processing exam database

Note: this code still needs to be significantly refactored, but it does seem to be useful for generating exams from a database of questions and answers.

```ruby
require 'json'
# https://hackhands.com/ruby-read-json-file-hash/
# note: if you don't have json, you need to do: gem install json
# however, since json is part of stdlib, you really should have it already
# http://ruby-doc.org/stdlib-2.4.0/libdoc/json/rdoc/JSON.html

class NilClass
  def size
    0
  end
end

class Array
  def rest
    self[1..-1]
  end
  def second
    self[1]
  end
end
```
class String
  def multiline(width)
    array = self.split
    rebuild array, width
  end

  private def rebuild(array, width)
    if array.size === 0 then
      []
    elsif array.size === 1 then
      array
    elsif array.first.size <= width then
      if (array.first + array.second).size <= width then
        rebuild ([array.first + " " + array.second] + array.rest.rest),
      width
      else
        [array.first] + (rebuild array.rest, width)
      end
    else
      [ array.first ] + (rebuild array.rest, width)
    end
  end
end

class ExamDatabase
  attr_reader :items
  def initialize(exam_database_file)
    file_contents = File.read exam_database_file
    @file_size = file_contents.size
    @database = JSON.parse file_contents
    @items = extract @database
  end

  def dump(formatter)
    dump_array @items, formatter
  end

  private def extract(database)
    items = @database["questions"].compact
    @items = items.map { |item| Item.new(item) }
  end

  def dump_array(array, formatter, filter=:+both)
    count = 0
    array.each do |item|
      count = count + 1
      item.dump count, formatter, filter
    end
  end
end

class Chapters
  attr_reader :chapters, :distribution
  def initialize(items)
@chapters = (items.map { | item | item.source }).uniq
@distribution = build_distribution(items)
@chapterize = build_chapterization(items)

end

def size
  result = 0
  @chapters.map { | chapter | result += @chapterize[chapter].size }
  result
end

def sample_round prng
  remaining_items = Array.new
  items_selected = Array.new
  @chapters.map do | chapter |
    current_items = @chapterize[chapter]
    current_sample = current_items.sample random: prng
    items_selected = items_selected << current_sample
    current_items = current_items.select { | item | item !=
      current_sample }
    remaining_items = remaining_items.concat(current_items)
  end
  return remaining_items, items_selected
end

private def build_distribution(items)
  distribution = Hash.new
  @chapters.map { | chapter | distribution[chapter] = 0 }
  items.map { | item | distribution[item.source] += 1 }
  distribution
end

private def build_chapterization(items)
  chapterize = Hash.new
  @chapters.map { | chapter | chapterize[chapter] = Array.new }
  items.map do | item |
    source = item.source
    chapterize[source] = chapterize[source] << item
  end
  chapterize
end

end

class Item
  def initialize(item_parts)
    @question = item_parts["question"]
    if item_parts.key? "lquestion" then
      @lquestion = item_parts["lquestion"]
    else
      @lquestion = @question
    end
    @answer = item_parts["answer"]
    if item_parts.key? "lanswer" then
      @lanswer = item_parts["lanswer"]
    else
      @lanswer = @answer
    end
  end
end
end
@source = item_parts['source']
end

def dump(count, formatter, filter)
    formatter.skip_line
    if filter == :both then
        formatter.dump_both self, filter, count
    elsif filter == :question then
        formatter.dump_question self, filter, count
    elsif filter == :answer then
        formatter.dump_answer self, filter, count
    else
        puts "Unknown filter: " + filter.to_s
    end
end
end

class PlainFormatter
    def initialize(width)
        @width = width
    end
    def prologue
        puts <<-END_OF_STRING
FINAL EXAM: CS97622A
NAME AS APPEARS ON STUDENT ID:
STUDENT ID NUMBER:
UWO/CONFLUENCE USER NAME:
REMINDERS:
** (from course outline) The exam will be closed book, closed notes, with no electronic devices allowed, with particular reference to any electronic devices that are capable of communication and/or storing information.

** Write neatly. If the marker can't read it, it is wrong.

** This exam shouldn't take long to write. On the other hand, time will pass. It is a 2 hour exam with 50 questions. If you complete a question every 2 minutes, you will still have 20 minutes at the end to double check that everything is in order.

** While you are not allowed to open the exam booklet until the proctor says you can, you can fill out the information on the cover page. You should also get out your student id and make sure your pencils and pens are in order. If you need to get something out of your jacket or knapsack once the exam has started, raise your hand and wait til a proctor comes to you to oversee the matter.

END_OF_STRING
end
def epilogue
end

def page_break
  puts "\f"
end

def skip_line
  puts ""
end

def dump_both(item, filter, count)
  question = "QUESTION " + count.to_s + ": " + item.question
  @width
  question.each { |line| puts line }
  dump_answer item, filter, count
end

def dump_question(item, filter, count)
  question = "QUESTION " + count.to_s + ": " + item.question
  @width
  question.each { |line| puts line }]
  puts "ANSWER="
end

def dump_answer(item, filter, count)
  if item.answer.instance_of? String then
    dump_answer_string item.answer, filter, count
  elsif item.answer.instance_of? Array then
    item.answer.each { |answer| dump_answer_string answer, filter, count }
  else
    puts "UNKNOWN ANSWER FORMAT"
  end
end

def dump_answer_string(answer, filter, count)
  if filter === :answer then
    answer_prefix = "ANSWER " + count.to_s + ": "
  else
    answer_prefix = "ANSWER=
  end
  puts answer_prefix + answer
end

def dump_configuration(config, items, possible_exam)
  config.dump
  chapters = Chapters.new(items).chapters.sort.to_s.multiline(@width)
  chapters.each { |line| puts line }
  chapters = Chapters.new(possible_exam).chapters.sort.to_s.multiline(@width)
  chapters.each { |line| puts line }
  puts "---------------"
end

class LatexFormatter
  def initialize
    @width = 72
  end
end
\documentclass{exam}
\begin{document}
FINAL EXAM: CS9622A \\
\newline
\newline
\newline
\newline
NAME AS APPEARS ON STUDENT ID: \\
\newline
STUDENT ID NUMBER: \\
\newline
UWO/CONFLUENCE USER NAME: \\
\newline
REMINDERS:
\begin{enumerate}
\item (from course outline) The exam will be closed book, closed notes, with no electronic devices allowed, with particular reference to any electronic devices that are capable of communication and/or storing information.
\item Write neatly. If the marker can't read it, it is wrong.
\item This exam shouldn't take long to write. On the other hand, time will pass. It is a 2 hour quiz with 50 questions. If you complete a question every 2 minutes, you will still have 20 minutes at the end to double check that everything is in order.
\item While you are not allowed to open the exam booklet until the proctor says you can, you can fill out the information on the cover page. You should also get out your student id and make sure your pencils and pens are in order. If you need to get something out of your jacket or knapsack once the exam has started, raise your hand and wait till a proctor comes to you to oversee the matter.
\end{enumerate}
\newpage
\begin{enumerate}
\end{enumerate}
\end{document}
puts "end{itemize}"
end

def dump_question(item, filter, count)
    question = "item " + item.lquestion
    puts question + "newline"
    puts "\begin{itemize}
end

def dump_answer(item, filter, count)
    if item.lanswer.instance_of? String then
        dump_answer_string item.lanswer, filter, count
    elsif item.answer.instance_of? Array then
        puts "\item begin\itemize" if filter != :both
        item.lanswer.each { |answer| dump_answer_string answer, filter, count }
    puts "end\itemize" if filter != :both
    else
        puts "UNKNOWN ANSWER FORMAT"
    end
end

def dump_answer_string(answer, filter, count)
    if filter === :answer then
        answer_prefix = "item "
    else
        answer_prefix = "item "
    end
    puts answer_prefix + answer
end

def dump_configuration(config, items, possible_exam)
    puts "\begin{verbatim}
config.dump
chapters = Chapters.new(items).chapters.sort.to_s.multiline(@width)
chapters.each { |line| puts line }
chapters = Chapters.new(possible_exam).chapters.sort.to_s.multiline
chapters.each { |line| puts line }
puts "\end{verbatim}"
end

class ExamConfig
    attr_reader :exam_database_file, :exam_format, :dump_database,
    :question_count, :create_exam, :answer_key, :sample_prng,
    :shuffle_prng, :exam_formatter

    def initialize(arg_list)
        if arg_list.any? { |value| /^database=/ =~ value } then
            database_arg = arg_list.find { |value| /^database=/ =~ value }
            @exam_database_file = database_arg.sub "database=set"
        else
            @exam_database_file = "examdatabase.json"
        end
        if arg_list.any? { |value| /^width=/ =~ value } then
            width_arg = arg_list.find { |value| /^width=/ =~ value }
        end
    end
end
@line_width = (width_arg.sub "width="", ").to_i
else
  # original line width limit for American teletypewriters
  # https://en.wikipedia.org/wiki/Characters_per_line
  @line_width = 72
end
if arg_list.any? { |value| /^question_count=/ =~ value } then
  question_count_arg = arg_list.find { |value| /^question_count=/ =~ value }
@question_count = (question_count_arg.sub "question_count="", ").to_i
else
  @question_count = 50
end
if arg_list.any? { |value| /^shuffle_seed=/ =~ value } then
  shuffle_seed_arg = arg_list.find { |value| /^shuffle_seed=/ =~ value }
@shuffle_seed = (shuffle_seed_arg.sub "shuffle_seed="", ").to_i
@shuffle_prng = Random.new @shuffle_seed
else
  @shuffle_seed = 654
  @shuffle_prng = Random.new @shuffle_seed
end
if arg_list.any? { |value| /^sample_seed=/ =~ value } then
  sample_seed_arg = arg_list.find { |value| /^sample_seed=/ =~ value }
@sample_seed = (sample_seed_arg.sub "sample_seed="", ").to_i
@sample_prng = Random.new @sample_seed
else
  @sample_seed = 6504
  @sample_prng = Random.new @sample_seed
end
if arg_list.include?("surprise_me") then
  @sample_seed = "unknown"
  @shuffle_seed = "unknown"
  @shuffle_prng = Random.new
  @sample_prng = Random.new
end
if arg_list.include?("latex") then
  @exam_format = "latex"
  @exam_formatter = LatexFormatter.new
else
  @exam_format = "plain"
  @exam_formatter = PlainFormatter.new @line_width
end
if arg_list.include?("dump_database") then
  @dump_database = true
else
  @dump_database = false
end
if arg_list.include?("create_exam") then
  @create_exam = true
else
  @create_exam = false
end
if arg_list.include?("answer_key") then
  @answer_key = true
else
  @answer_key = false
end
if arg_list.include?("dump") then
dump
end
def dump
  puts "exam_database_file= " + @exam_database_file
  puts "exam_format= " + @exam_format
  puts "dump_database= " + @dump_database.to_s
  puts "line_width= " + @line_width.to_s
  puts "question_count= " + @question_count.to_s
  puts "create_exam= " + @create_exam.to_s
  puts "answer_key= " + @answer_key.to_s
  puts "sample_seed= " + @sample_seed.to_s
  puts "shuffle_seed= " + @shuffle_seed.to_s
end
def print_chapter_distribution(items)
  the_chapters = Chapters.new items
  the_chapters.chapters.sort.each do | chapter |
    puts chapter + " : " + the_chapters.distribution[chapter].to_s
  end
end
def build_exam(items, config)
  exam = Array.new
  count = config.question_count

  chapters = Chapters.new(items)
  if count >= chapters.chapters.size then
    items, round1 = chapters.sample_round config.sample_prng
    count = count - round1.size
    exam = exam.concat round1
  end

  chapters = Chapters.new(items)
  if count >= chapters.chapters.size then
    items, round2 = chapters.sample_round config.sample_prng
    count = count - round2.size
    exam = exam.concat round2
  end

  exam = exam.concat items.sample(count, random: config.sample_prng)
  exam = exam.shuffle(random: config.shuffle_prng)
  exam = exam.shuffle(random: config.shuffle_prng)
  exam = exam.shuffle(random: config.shuffle_prng)
end
config = ExamConfig.new ARGV
exam_database = ExamDatabase.new config.exam_database_file
if config.dump_database then
  config.exam_formatter.prologue
  exam_database.dump config.exam_formatter
  config.exam_formatter.epilogue
end
if config.create_exam
  config.exam_formatter.prologue
  possible_exam = build_exam exam_database.items, config
  exam_database.dump_array possible_exam, config.exam_formatter
  config.exam_formatter.epilogue
end
if config.answer_key
  config.exam_formatter.prologue
  possible_exam = build_exam exam_database.items, config
  exam_database.dump_array possible_exam, config.exam_formatter, :question
  config.exam_formatter.page_break
  config.exam_formatter.dump_configuration config, exam_database.items, possible_exam
  exam_database.dump_array possible_exam, config.exam_formatter, :answer
  config.exam_formatter.epilogue
end

Script used to build exams from database (runs the ruby program with various parameter settings and invokes pdflatex)

#!/bin/csh -f
# plain text and latex dumps of exam database
ruby makeExam.rb plain dump_database > examdatabase.plain
ruby makeExam.rb latex dump_database > examdatabase.tex
pdflatex examdatabase.tex
# plain text and latex dumps of 200 question exam (whole database)
answer_key format
ruby makeExam.rb plain sample_seed=222 shuffle_seed=2345 answer_key
question_count=200 > examdatabaseAK.plain
ruby makeExam.rb latex sample_seed=222 shuffle_seed=2345 answer_key
question_count=200 > examdatabaseAK.tex
pdflatex examdatabaseAK.tex
# plain text and latex dumps of a sample 50 question exam, answer_key format
ruby makeExam.rb plain sample_seed=222 shuffle_seed=245 answer_key
question_count=50 > examdatabaseAK50.plain
ruby makeExam.rb latex sample_seed=222 shuffle_seed=245 answer_key
question_count=50 > examdatabaseAK50.tex
pdflatex examdatabaseAK50.tex

parameters:
- dump_database – list questions followed immediately by their answers
- plain – give plain text version of exam
- latex – give latex versions of exam
- sample_seed – seed random number generator with this seed when choosing which questions to use from the database
- shuffle_seed – seed random number generator with this seed when choosing the order the questions will appear in in the exam.
- answer_key – list questions without answers and then list answers without questions (both lists use same ordering)
- question_count – number of questions that will be listed (unless end of database is encountered first)

Exam database for questions covering the first 10 classes

```
{
    "questionTemplate": {
        "question": "What is the answer?",
        "lquestion": "latex reformatting for questions that needed it, absent if not needed",
        "answer": "That which was asked for",
        "lanswer": "latex reformatting for answers that needed it, absent if not needed",
        "note": "optional note",
        "source": "C num -- number of class where this material was
```
"description": "questions from which to construct the Fall 2017 final exam in CS9622a",
"questions": [
  {
    "question": "ANSWER is the name of the federal payroll system that has regularly appeared in the news since February 2016 due to federal employees not getting paid",
    "answer": "Phoenix Pay",
    "note": "optional note",
    "source": "C1"
  },
  {
    "question": "CS9622a is primarily about the handling of non-functional requirements with particular focus on requirements associated with safety, accessibility, and ANSWER",
    "answer": "sustainability",
    "note": "optional note",
    "source": "C1"
  },
  {
    "question": "CS9622 focusses on requirements arising from how software components fit into larger ANSWER",
    "answer": "socio-technical systems",
    "note": "optional note",
    "source": "C1"
  },
  {
    "question": "A system is defined as a set of parts connected to one another to ANSWER",
    "answer": "achieve a certain goal",
    "note": "optional note",
    "source": "C2"
  },
  {
    "question": "A feedback loop that is self-correcting is called ANSWER",
    "answer": "a balancing feedback loop",
    "note": "optional note",
    "source": "C2"
  },
  {
    "question": "ANSWER is a quantity that can increase over time as a result of an in-flow and decrease over time as the result of an out-flow",
    "answer": "A stock",
    "note": "optional note",
    "source": "C2"
  },
  {
    "question": "A feedback loop that gains strength with each cycle is called ANSWER",
    "answer": "a reinforcing feedback loop"}
Software requirements are a bridge between the real world and ANSWER,

Non-functional requirements are requirements that define ANSWER,

The principle cause of safety-related software errors that persist till integration and system testing is ANSWER,

The difficulties for requirements engineering are that the customer often doesn't know what system they want or can't agree on what they want and ANSWER,

Customer politics are a problem for requirements engineering because ANSWER,

Comparing safety concerns with security concerns, safety experts are trying to prevent loss due to ANSWER,

Comparing safety concerns with security concerns, security experts are trying to prevent loss due to ANSWER,
Leveson's model of accident causality is called STAMP, and Leveson sees safety and security as resulting from imposing constraints on the system behavior. In addition to having a model of accident causality, Leveson also developed a technique for using that model to do hazard analysis. The technique was called STPA. In June 2017, a Miami federal judge ruled that Winn-Dixie, a grocery store, violated the Americans with Disabilities Act due to inaccessibility of their web site for blind users. Universal design, inclusive design, and design-for-all are defined as a framework for the design of places, things, information, communications, and policy that focuses on the user, on the widest range of people operating in the widest range of situations without special or separate design. Leveson criticizes traditional hazard analysis methods like Fault Tree Analysis and Failure Modes and Effects Analysis as being designed to identify component failure accidents and not component interaction accidents.
"question": "Since software seldom degrades like hardware, the focus on component failure accidents means that Leveson finds traditional hazard analysis methods not appropriate for ANSWER",
"answer": "software",
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson defines hazards as ANSWER",
"answer": "potential causes of loss",
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson defines accidents as ANSWER",
"answer": ["unplanned events that cause loss","undesired events that cause loss"],
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson says accidents result from ANSWER",
"answer": "a lack of enforcement of safety constraints",
"note": "optional note",
"source": "C3"
},
{
"question": "Guessing at passwords over ssh by using information about how long it takes to type one character after another by a particular typist is an example of a ANSWER attack.",
"answer": "timing",
"note": "optional note",
"source": "C3"
},
{
"question": "The paper on guessing passwords typed over ssh was an example of how ANSWER is not so much a property of a program (here ssh), but rather of the system including the program, the user, and the supporting hardware.",
"answer": "security",
"note": "optional note",
"source": "C3"
},
{
"question": "The paper Anayzing the Harmful Effect of God Class Refactoring on Power Consumption illustrated the possibility of a conflict between various non-functional requirements, in this case maintainability and ANSWER",
"answer": "sustainability",
"note": "optional note",
"source": "C3"
}
Causal linkages, feedback loops, rates and levels, and structural behavioral relationships are four important aspects of system dynamics.

Feedback loops are defined as closed changes of causal relations.

Important aspects of causal linkages are that they can be non-linear and have time delays.

A positive feedback loop in isolation will generally generate exponential growth.

A negative feedback loop drives a system toward its goal level with diminishing return.

In a systematic literature review, we start by defining the research questions we want to investigate.

An important issue in a systematic literature review is how one chooses the literature to be reviewed.

The paper Integration Between Requirements Engineering and Safety Analysis: A Systematic Literature Review advanced the field by organizing various concepts encountered into...
{  
  "question": "Automatically magnifying web pages to eliminate side margins was an example of ANSWER",
  "answer": "opportunistic accessibility",
  "note": "optional note",
  "source": "C4"
},

{  
  "question": "In representational technology, the system is viewed as representing the designer's view on the knowledge the system reflects. In performative design, the system is viewed as reflecting the knowledge of ANSWER",
  "answer": "the user",
  "note": "optional note",
  "source": "C4"
},

{  
  "question": "One of the underlying themes of Requirements Engineering: A Roadmap was that non-functional goals are generally viewed as qualitative goals and that improvement in how they are managed requires converting them to ANSWER",
  "answer": "quantitative goals",
  "note": "optional note",
  "source": "C5"
},

{  
  "question": "The paper Measuring Software Sustainability distinguishes maintainability from sustainability by saying that maintainability is about the software, whereas sustainability is about ANSWER",
  "answer": "the whole software evolution process",
  "note": "optional note",
  "source": "C5"
},

{  
  "question": "The key point of Measuring Software Sustainability is that we can measure a qualitative requirement on a software process by focussing on ANSWER devoted to ensuring that the requirement holds across the lifetime of the software.",
  "answer": "the amount of time",
  "note": "optional note",
  "source": "C5"
},

{  
  "question": "The paper A Systematic Literature Review of Software Sustainability Measures points out that although people often associate sustainability with environmental impact, one also needs to address negative impacts on human beings, society, and ANSWER",
  "answer": "the economy",
  "note": "optional note",
  "source": "C5"}
The paper A Systematic Literature Review of Software Sustainability Measures points out that although people often associate sustainability with user usage of a software system, one also needs to consider the system's deployment and development.

In the thesis Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities, the qualitative concept of accessibility was quantified by a weighted counting of violations of Web Content Accessibility Guidelines that could be checked by a computer program.

In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, the traditional ways of measuring quality of web applications were the PULSE metrics: Page views, Uptime, Latency, Seven-day active, and earnings.

In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, the authors propose measuring quality of web applications with Heart metrics: Happiness, Engagement, Adoption, Retention, and earnings.

Happiness is measured by user surveys.

Engagement, Adoption, and Retention metrics give us a finer tuned view of the application than the traditional PULSE metrics.
ANSWER metrics of the PULSE system.

    "answer": "Seven-day active users",
    "note": "optional note",
    "source": "C5"
},

{
    "question": "Systems theorist distinguish three kinds of systems -- those with organized simplicity, those with disorganized complexity, and the third kind which they are most interesting in, those with ANSWER",
    "answer": "organized complexity",
    "note": "optional note",
    "source": "C6"
},

{
    "question": "When analyzing systems with disorganized complexity, the appropriate methodology is ANSWER",
    "answer": "statistics",
    "note": "optional note",
    "source": "C6"
},

{
    "question": "The concept of emergence is the idea that at a given level of the hierarchy of descriptions of the system, some properties characteristic of that level are ANSWER",
    "answer": ["irreducible","cannot be reduced to properties at lower levels"],
    "note": "optional note",
    "source": "C6"
},

{
    "question": "The assumption that the tester routinely knows what the correct answer for a test is is called ANSWER",
    "answer": "the oracle assumption",
    "note": "optional note",
    "source": "C6"
},

{
    "question": "Techniques for testing software that doesn't have specifications is called ANSWER",
    "answer": "exploratory testing",
    "note": "optional note",
    "source": "C6"
},

{
    "question": "James Robertson presented an approach to how to make requirements measurable called ANSWER",
    "answer": "the fit criteria",
    "note": "optional note",
    "source": "C6"
},

{
    "question": "The Erlang language for distributed programming handles failures by ANSWER",
    "answer": "the fit criteria",
    "note": "optional note",
    "source": "C6"
}
"answer": "having an observing process that detects and handles failures monitor the working process",
"note": "optional note",
"source": "C6"
},

{ "question": "The paper One Bit Flips, On Cloud Flops discussed a security exploit that showed how the underlying weakness of the hardware design could undermine the security of a cloud user. The attack was called row-hammer and specifically addressed weaknesses in ANSWER",
"answer": "dynamic ram memory",
"note": "optional note",
"source": "C6"
},

{ "question": "The current state of the art for handling compliance issues is internal IT auditors and ANSWER",
"answer": "third party certifying agencies",
"note": "optional note",
"source": "C7"
},

{ "question": "Software regulation documents are generally not designed for programmers, but rather for ANSWER",
"answer": "lawyers",
"note": "optional note",
"source": "C7"
},

{ "question": "The GreenOracle paper claimed that reasonable energy usage estimates for a system could be made based on cpu utilization and ANSWER",
"answer": "system calls",
"note": "optional note",
"source": "C7"
},

{ "question": "The Building a Safety Architecture System paper presented a collection of software patterns aimed at handling failure of ANSWER",
"answer": "a few components of the system",
"note": "optional note",
"source": "C7"
},

{ "question": "Safety requirements, like other requirements, are determined by the system ANSWER",
"answer": "stakeholders",
"note": "optional note",
"source": "C7"
},

{ "question": "STPA begins at the ANSWER level of the hierarchical
description of the system",
   "answer": "top",
   "note": "optional note",
   "source": "C7"
},
{
   "question": "The paper Goal-Based Requirements Analysis focussed on the issue of ANSWER",
   "answer": "how do you identify goals",
   "note": "optional note",
   "source": "C8"
},
{
   "question": "The Goal Analysis approach to requirements identifies, organizes, and classifies goals based on ANSWER",
   "answer": "gathered documentation",
   "note": "optional note",
   "source": "C8"
},
{
   "question": "The Goal Analysis approach to requirements is concerned not only with identifying goals, but also with identifying agents and ANSWER",
   "answer": "stakeholders",
   "note": "optional note",
   "source": "C8"
},
{
   "question": "The Privacy Impacts of IoT Devices paper based its notion of privacy on the US Federal Trade Commission's fair information practice principles. It analyzed the Samsung SmartTV device based on ANSWER",
   "answer": "Samsung's privacy policies",
   "note": "optional note",
   "source": "C8"
},
{
   "question": "Step 1 of STPA is to identify the potential for ANSWER that could lead to a hazardous state. Step 2 of STPA is then to determine how things identified in Step 1 could occur.",
   "answer": ["inadequate control","inadequate enforcement of safety constraints"],
   "note": "optional note",
   "source": "C8"
},
{
   "question": "One way to verify the STPA analysis was done correctly is to revisit it after ANSWER",
   "answer": "accidents and incidents",
   "note": "optional note",
   "source": "C8"
},

"question": "As the system design evolves, the STPA analysis must be ANSWER",
"answer": "re-evaluated and updated",
"note": "optional note",
"source": "C8"
},
{
"question": "The difference between a human component in a system and an automated controller is that while both require internal models of the components they are controlling, human controllers also require ANSWER",
"answer": "internal models of components that they indirectly control",
"note": "optional note",
"source": "C8"
},
{
"question": "The paper One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques considered the techniques: exploratory manual penetration testing, static analysis, automated penetration testing, and systematic manual penetration testing. They measured the efficiencies of these techniques with the metric ANSWER",
"answer": "vulnerabilities found per hour",
"note": "optional note",
"source": "C9"
},
{
"question": "The paper One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques considered the techniques: exploratory manual penetration testing, static analysis, automated penetration testing, and systematic manual penetration testing. They found systematic manual penetration testing most efficient, but recommended that it be combined with static analysis because ANSWER",
"answer": "the two methods found different kinds of vulnerabilities",
"note": "optional note",
"source": "C9"
},
{
"question": "In the paper Towards Enhanced Usability of IT Security Mechanisms, the main issue is that in order for a security mechanism to be successful, users must find it ANSWER",
"answer": "usable",
"note": "optional note",
"source": "C9"
},
{
"question": "In the paper Towards Enhanced Usability of IT Security Mechanisms, an example of a security method explored was ANSWER",
"answer": 
["email encryption", "GPGMail", "forced updates", "captchas", "HTTPS certificate validation in common browsers"],
"note": "optional note",
"source": "C9"}
In Exploring Language Technologies to Provide Support to WCAG 2.0 and E2R guidelines, the reason we are interested in language tools is because these accessibility guidelines require \textit{web pages be readable}, \textit{web page readability}, \textit{web pages be understandable}, \textit{web page understandability},
\begin{itemize}
  \item \textit{web pages be readable}
  \item \textit{web page readability}
  \item \textit{web pages be understandable}
  \item \textit{web page understandability}
\end{itemize}
\textit{note}: \textit{optional note},
\textit{source}: \textit{C9}

Architectural sustainability is defined as the set of factors that promote an architecture's \textit{stability and longevity during system evolution},
\textit{note}: \textit{optional note},
\textit{source}: \textit{C9}

One aspect of software architectures is the documented design decisions from which they were constructed. One way of measuring the sustainability of such architectural knowledge is \textit{ripple effect metrics}, \textit{instability}, \textit{change proneness}, \textit{size of captured architectural knowledge},
\textit{note}: \textit{optional note},
\textit{source}: \textit{C9}

The paper Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems used STPA on incident reports from the FDA Maude database to create hazard scenarios. These scenarios were then tested using the technique of \textit{fault injection},
\textit{note}: \textit{optional note},
\textit{source}: \textit{C10}

In the paper Prejudices, Memories, Expectations, and Confidence Influence Experienced Accessibility on the Web, the authors concluded that experienced accessibility of a web site was significantly influenced by \textit{past experiences}, \textit{prejudices}, \textit{expectations}, \textit{confidence},
\textit{note}: \textit{optional note},
\textit{source}: \textit{C10}

In the paper Quality Requirements in Agile as a Knowledge Management Problem, More than Just-in-Time, the authors claim that requirements like safety and security requirements requires an explicit knowledge management framework. Such a framework tracks requirements knowledge, attack knowledge, and a long list of requirements to detect \textit{ripple effect metrics}, \textit{instability}, \textit{change proneness}, \textit{size of captured architectural knowledge}. 

\textit{note}: \textit{optional note},
\textit{source}: \textit{C9}

\textit{source}: \textit{C10}

\textit{note}: \textit{optional note},
\textit{source}: \textit{C10}

\textit{note}: \textit{optional note},
\textit{source}: \textit{C10}

\textit{note}: \textit{optional note},
\textit{source}: \textit{C10}
In the paper Quality Requirements in Agile as a Knowledge Management Problem, More than Just-in-Time, the authors envision a change of the Agile development of secure systems so that instead of just being an interaction between customer and developer, there would also be a security expert involved.
First 10 classes, 50 randomly chosen questions with answer key

Questions and Answers for All Eleven Classes pdf

database of Questions and answers for all eleven classes
{  
  "questiontemplate": {  
    "question": "What is the answer?",  
    "lquestion": "latex reformatting for questions that needed it, absent if not needed",  
    "answer": "That which was asked for",  
    "lanswer": "latex reformatting for answers that needed it, absent if not needed",  
    "note": "optional note",  
    "source": "C num -- number of class where this material was covered"  
  },  
  "description": "questions from which to construct the Fall 2017 final exam in CS9622a",  
  "questions": [  
    {  
      "question": "ANSWER is the name of the federal payroll system that has regularly appeared in the news since February 2016 due to federal employees not getting paid",  
      "answer": "Phoenix Pay",  
      "note": "optional note",  
      "source": "C1"  
    },  
    {  
      "question": "CS9622a is primarily about the handling of non-functional requirements with particular focus on requirements associated with safety, accessibility, and ANSWER",  
      "answer": "sustainability",  
      "note": "optional note",  
      "source": "C1"  
    },  
    {  
      "question": "CS9622 focusses on requirements arising from how software components fit into larger ANSWER",  
      "answer": "socio-technical systems",  
      "note": "optional note",  
      "source": "C1"  
    }  
  ]}
A system is defined as a set of parts connected to one another to **achieve a certain goal**.

A feedback loop that is self-correcting is called **a balancing feedback loop**.

**A stock** is a quantity that can increase over time as a result of an in-flow and decrease over time as the result of an out-flow.

A feedback loop that gains strength with each cycle is called **a reinforcing feedback loop**.

Software requirements are a bridge between the real world and **the software system**.

Non-functional requirements are requirements that define **some observable characteristic of the system**, **a quality of the system**, **when there are two answers, you only have to know one of them**.

The principle cause of safety-related software errors that persist till integration and system testing is **getting the requirements wrong**.
"question": "The difficulties for requirements engineering are that the customer often doesn't know what system they want or can't agree on what they want and ANSWER",
"answer": ["the business changes","the system environment changes"],
"note": "optional note",
"source": "C2"
},

"question": "Customer politics are a problem for requirements engineering because ANSWER",
"answer": "the customer often can't tell the engineer what they really want",
"note": "optional note",
"source": "C2"
},

"question": "Comparing safety concerns with security concerns, safety experts are trying to prevent loss due to ANSWER",
"answer": "unintentional actions of user",
"note": "optional note",
"source": "C2"
},

"question": "Comparing safety concerns with security concerns, security experts are trying to prevent loss due to ANSWER",
"answer": "intentional actions of user",
"note": "optional note",
"source": "C2"
},

"question": "Leveson's model of accident causality is called ANSWER",
"answer": "STAMP",
"note": "optional note",
"source": "C2"
},

"question": "Leveson sees safety and security as resulting from imposing ANSWER on the system behavior.",
"answer": "constraints",
"note": "optional note",
"source": "C2"
},

"question": "In addition to having a model of accident causality, Leveson also developed a technique for using that model to do hazard analysis. The technique was called ANSWER",
"answer": "STPA",
"note": "optional note",
"source": "C2"
"question": "In June 2017, a Miami federal judge ruled that ANSWER violated the Americans with Disabilities Act due to inaccessibility of their web site for blind users",
"answer": ["Winn-Dixie","a grocery store"],
"note": "optional note",
"source": "C2"
},
{
"question": "ANSWER is defined as a framework for the design of places, things, information, communications, and policy that focuses on the user, on the widest range of people operating in the widest range of situations without special or separate design.",
"answer": ["universal design", "inclusive design", "design-for-all"],
"note": "optional note",
"source": "C2"
},
{
"question": "Leveson criticizes traditional hazard analysis methods like Fault Tree Analysis and Failure Modes and Effects Analysis as being designed to identify component failure accidents and not ANSWER",
"answer": "component interaction accidents",
"note": "optional note",
"source": "C3"
},
{
"question": "Since software seldom degrades like hardware, the focus on component failure accidents means that Leveson finds traditional hazard analysis methods not appropriate for ANSWER",
"answer": "software",
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson defines hazards as ANSWER",
"answer": "potential causes of loss",
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson defines accidents as ANSWER",
"answer": ["unplanned events that cause loss","undesired events that cause loss"],
"note": "optional note",
"source": "C3"
},
{
"question": "Leveson says accidents result from ANSWER",
"answer": "a lack of enforcement of safety constraints",
"note": "optional note",
"source": "C3"
}
"question": "Guessing at passwords over ssh by using information about how long it takes to type one character after another by a particular typist is an example of a **ANSWER** attack.",
"answer": "timing",
"note": "optional note",
"source": "C3"
},
{
"question": "The paper on guessing passwords typed over ssh was an example of how **ANSWER** is not so much a property of a program (here ssh), but rather of the system including the program, the user, and the supporting hardware.",
"answer": "security",
"note": "optional note",
"source": "C3"
},
{
"question": "The paper Analyzing the Harmful Effect of God Class Refactoring on Power Consumption illustrated the possibility of a conflict between various non-functional requirements, in this case maintainability and **ANSWER**",
"answer": "sustainability",
"note": "optional note",
"source": "C3"
},
{
"question": "Causal linkages, feedback loops, rates and levels, and structural behavioral relationships are four important aspects of **ANSWER**",
"answer": "system dynamics",
"note": "optional note",
"source": "C4"
},
{
"question": "Feedback loops are defined as closed changes of **ANSWER**",
"answer": "causal relations",
"note": "optional note",
"source": "C4"
},
{
"question": "Important aspects of causal linkages are that they can be non-linear and have **ANSWER**",
"answer": "time delays",
"note": "optional note",
"source": "C4"
},
{
"question": "A positive feedback loop in isolation will generally generate **ANSWER**",
"answer": "exponential growth",
"note": "optional note",
"source": "C4"
A negative feedback loop drives a system toward its goal level with diminishing return.

In a systematic literature review, we start by defining the questions we want to investigate.

An important issue in a systematic literature review is how one chooses the literature to be reviewed.

The paper Integration Between Requirements Engineering and Safety Analysis: A Systematic Literature Review advanced the field by organizing various concepts encountered into taxonomies.

Automatically magnifying web pages to eliminate side margins was an example of opportunistic accessibility.

In representational technology, the system is viewed as representing the designer's view on the knowledge the system reflects. In performative design, the system is viewed as reflecting the knowledge of the user.

One of the underlying themes of Requirements Engineering: A Roadmap was that non-functional goals are generally viewed as qualitative goals and that improvement in how they are managed requires converting them to quantitative goals.
The paper Measuring Software Sustainability distinguishes maintainability from sustainability by saying that maintainability is about the software, whereas sustainability is about the whole software evolution process.

The key point of Measuring Software Sustainability is that we can measure a qualitative requirement on a software process by focusing on the amount of time devoted to ensuring that the requirement holds across the lifetime of the software.

The paper A Systematic Literature Review of Software Sustainability Measures points out that although people often associate sustainability with environmental impact, one also needs to address negative impacts on human beings, society, and the economy.

The paper A Systematic Literature Review of Software Sustainability Measures points out that although people often associate sustainability with user usage of a software system, one also needs to consider the system's deployment and development.

In the thesis Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities, the qualitative concept of accessibility was quantified by a weighted counting of violations of the Web Content Accessibility Guidelines that could be checked by a computer program.

In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, the traditional ways of measuring quality of web applications were the PULSE metrics: Page views, Uptime, Latency, Seven-day active, and Earnings.
"note": "an interesting question is what did the author mean by "earnings" here",
"source": "C5"
},

{ "question": "In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, the authors propose measuring quality of web applications with Heart metrics: Happiness, Engagement, Adoption, Retention, and ANSWER",
"answer": "Task Success",
"note": "optional note",
"source": "C5"
},

{ "question": "In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, Happiness is measured by ANSWER",
"answer": "user surveys",
"note": "optional note",
"source": "C5"
},

{ "question": "In Measuring the User Experience on a Large Scale: User-Centered Metrics for Web Applications, the Engagement, Adoption, and Retention metrics give us a finer tuned view of the application than the ANSWER metrics of the PULSE system.",
"answer": "Seven-day active users",
"note": "optional note",
"source": "C5"
},

{ "question": "Systems theorist distinguish three kinds of systems -- those with organized simplicity, those with disorganized complexity, and the third kind which they are most interesting in, those with ANSWER",
"answer": "organized complexity",
"note": "optional note",
"source": "C6"
},

{ "question": "When analyzing systems with disorganized complexity, the appropriate methodology is ANSWER",
"answer": "statistics",
"note": "optional note",
"source": "C6"
},

{ "question": "The concept of emergence is the idea that at a given level of the hierarchy of descriptions of the system, some properties characteristic of that level are ANSWER",
"answer": ["irreducible","cannot be reduced to properties at lower levels"],
"note": "optional note",
"source": "C6"}
The assumption that the tester routinely knows what the correct answer for a test is is called **the oracle assumption**.

Techniques for testing software that doesn't have specifications is called **exploratory testing**.

James Robertson presented an approach to how to make requirements measurable called **the fit criteria**.

The Erlang language for distributed programming handles failures by **having an observing process that detects and handles failures monitor the working process**.

The paper One Bit Flips, On Cloud Flops discussed a security exploit that showed how the underlying weakness of the hardware design could undermine the security of a cloud user. The attack was called row-hammer and specifically addressed weaknesses in **dynamic ram memory**.

The current state of the art for handling compliance issues is internal IT auditors and **third party certifying agencies**.

Software regulation documents are generally not designed for programmers, but rather for **lawyers**.
The GreenOracle paper claimed that reasonable energy usage estimates for a system could be made based on CPU utilization and **answer**: system calls,

The Building a Safety Architecture System paper presented a collection of software patterns aimed at handling failure of **answer**: a few components of the system,

Safety requirements, like other requirements, are determined by the system **answer**: stakeholders,

STPA begins at the **answer**: top level of the hierarchical description of the system,

The paper Goal-Based Requirements Analysis focussed on the issue of **answer**: how do you identify goals,

The Goal Analysis approach to requirements identifies, organizes, and classifies goals based on **answer**: gathered documentation,

The Goal Analysis approach to requirements is concerned not only with identifying goals, but also with identifying agents and **answer**: stakeholders,
The Privacy Impacts of IoT Devices paper based its notion of privacy on the US Federal Trade Commission's fair information practice principles. It analyzed the Samsung SmartTV device based on Samsung's privacy policies.

Step 1 of STPA is to identify the potential for inadequate control that could lead to a hazardous state. Step 2 of STPA is then to determine how things identified in Step 1 could occur.

One way to verify the STPA analysis was done correctly is to revisit it after accidents and incidents.

As the system design evolves, the STPA analysis must be re-evaluated and updated.

The difference between a human component in a system and an automated controller is that while both require internal models of the components they are controlling, human controllers also require internal models of components that they indirectly control.

The paper One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques considered the techniques: exploratory manual penetration testing, static analysis, automated penetration testing, and systematic manual penetration testing. They measured the efficiencies of these techniques with the metric vulnerabilities found per hour.
The paper One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques considered the techniques: exploratory manual penetration testing, static analysis, automated penetration testing, and systematic manual penetration testing. They found systematic manual penetration testing most efficient, but recommended that it be combined with static analysis because the two methods found different kinds of vulnerabilities.

In the paper Towards Enhanced Usability of IT Security Mechanisms, the main issue is that in order for a security mechanism to be successful, users must find it usable.

In the paper Towards Enhanced Usability of IT Security Mechanisms, an example of a security method explored was email encryption, GPGMail, forced updates, captchas, HTTPS certificate validation in common browsers.

In Exploring Language Technologies to Provide Support to WCAG 2.0 and E2R guidelines, the reason we are interested in language tools is because these accessibility guidelines require web pages be readable, web page readability, web pages be understandable, web page understandability.

Architectural sustainability is defined as the set of factors that promote an architecture's stability and longevity during system evolution.

One aspect of software architectures is the documented design decisions from which they were constructed. One way of measuring the sustainability of such architectural knowledge is ripple effect metrics, instability, change proneness, size of captured architectural knowledge.
"question": "The paper Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems used STPA on incident reports from the FDA Maude database to create hazard scenarios. These scenarios were then tested using the technique of ANSWER",
"answer": "fault injection",
"note": "optional note",
"source": "C10"
},
{
"question": "In the paper Prejudices, Memories, Expectations, and Confidence Influence Experienced Accessibility on the Web, the authors concluded that experienced accessibility of a web site was significantly influenced by ANSWER and was not simply a property inherent in the web site itself.",
"answer": ["past experiences","prejudices", "expectations", "confidence"],
"note": "optional note",
"source": "C10"
},
{
"question": "In the paper Quality Requirements in Agile as a Knowledge Management Problem, More than Just-in-Time, the authors claim that requirements like safety and security requirements requires an explicit knowledge management framework. Such a framework tracks requirements knowledge, attack knowledge, and a long list of requirements to detect ANSWER",
"answer": "suspicious requirements",
"note": "optional note",
"source": "C10"
},
{
"question": "In the paper Quality Requirements in Agile as a Knowledge Management Problem, More than Just-in-Time, the authors envision a change of the Agile development of secure systems so that instead of just being an interaction between customer and developer, there would also be ANSWER involved."
"answer": "a security expert",
"note": "optional note",
"source": "C10"
},
{
"question": "ANSWER is a publically available usability measuring tool based on cognitive modelling to address the question of how long it would take an expert user to accomplish a task on the system."
"answer": "CogTool",
"note": "optional note",
"source": "C11"
},
{
"question": "A popular cognitive modelling system applied to usability analysis and designed at Carnegie Mellon is ANSWER"
"answer": ["ACT-R","SOAR"],
"note": "optional note"
Anderson's cognitive modelling system handles two kinds of knowledge, declaritive and procedural, chunks and production rules.

In Anderson's cognitive modelling system, choosing what knowledge is appropriate to a given situation is handled by Bayesian reasoning.

In the paper Measuring the Security Impacts of Password Policies Using Cognitive Behavioral Agent-Based Modelling, we see another approach to understanding the system where a non-functional requirement must hold, i.e., through simulation of humans interacting with the system using the technique of Agent-Based Modelling.
CS9622 Fall 2018 Version of Course

- Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability
- First Class: 10 Sept 2018

First Class: 10 Sept 2018


2. Issues in Requirements
   1. Eliciting Requirements from Stakeholders
   2. Negotiating the Expectations of Stakeholders into a Consistent Implementable Set of Requirements
   3. Refining Requirements for Designers of Expected System
   4. Tracing Connections between Requirements and Implemented System (said system, over time, will have to be adapted to new and changing requirements)
   5. Determining if Implemented System Meets Requirements

3. Issues in Non-Functional Requirements (sometimes referred to as Quality Requirements or the Illities)
   1. The non-functional requirements of most interest to this class are not properties of a piece of code, but rather properties of a system that includes software components.
   2. In order for the programmer to meet these requirements, thought must be given to how a software component will interact with the rest of the system.
   3. The relevant systems generally include hardware, software, and human components; and are sometimes referred to as socio-technical systems.
4. The systems of interest generally evolve over time as both the environment they are within changes and the human components learn from the behavior of the system and adapt their role within it to their view of how it works.
5. The disciplines of ‘systems engineering’, ‘systems design’, and ‘system dynamics’ play a major role in the discussion of how to address these sorts of requirements.

4. Safety (and Security)
   1. safety is about considering what could go wrong and incorporating components to monitor when things have gone wrong and take appropriate corrective action
   2. the stuff that could go wrong is referred to as ‘hazards’ and the checks done by the monitors are known as ‘safety constraints’
   3. current systems are very complicated and their failure is often the result of the interactions of components that are individually functioning ‘correctly’, but whose interaction violates some safety constraint of the system.
   4. a situation that would be traditionally analyzed as an ‘operator failure’ is often more usefully viewed as a ‘design failure’ of the system the operator was a part of.
   5. a method of interest for addressing these issues is STAMP: Systems-Theoretic Accident Model and Processes by Nancy Leveson (see Engineering a safer world: systems thinking applied to safety (book))

5. Accessibility (and Usability)
   1. there are guidelines, like WCAG, to prevent obvious errors like not putting alternative text on an image on a web page; but these don’t get at the fundamental issues of whether or not a system is convenient to use
   2. design of such systems should closely involve the people who will use them (or a representative sampling thereof)
   3. evaluation of accessibility and usability often involves qualitative research methods being applied to people using the system
   4. approaches to address these matters are often called ‘universal design’ or ‘inclusive design’. A good intro to these issues is the PhD Thesis Inclusive design of ICT: The challenge of diversity (U Oslo thesis) by Kristin Fuglerud.

6. Sustainability
   1. three views: creating systems that are sustainable; focussing on the energy consumption of software components of systems; and specifically maintaining software components of systems across the long term
   2. four aspects of sustainability: environmental, social, cultural, and economic
   3. while less developed than safety and accessibility, a good overview can be found in the PhD Thesis Software Engineering for Sustainability (thesis) by Birgit Penzenstadler

7. Structure of this Web Site
   1. Top level indexing of material relevant to the course
      1. Course Announcements
      2. Thinking about systems
      3. Understanding humans
      4. Overview of non-functional software requirements and requirements engineering
      5. From requirements to design
      6. How safety requirements figure into software systems
      7. How accessibility requirements figure into software systems
      8. How sustainability requirements figure into software systems
      9. non-functional requirements figure into software systems – Privacy, Reliability, Security, Usability, Learnability, Adaptability, and Technology Acceptance
      10. Writing papers in computer science
   2. The way the course developed in Fall 2017: CS9622 Fall 2017 version
   3. Material I have been looking at since Fall 2017: Lectures to an Empty Classroom
   4. A note on searching the computer science literature: Searching the Computer Science Literature

8. Your Upcoming Deadlines: (from Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability)
   1. 24 Sept 2018: in-class status report on upcoming Proposal for In Depth work
   2. 1 Oct 2018: Proposal for In Depth work due
   3. Quote from Second Class (Monday) 18 Sept 2017 (last year)
      1. Since a project proposal is due 2 Oct 2017 (see Course Outline for Computer Science 9622a: Nonfunctional Software Requirements: Safety, Accessibility, and Sustainability) , it is worth looking at Thesis Projects A Guide for Students in Computer Science and Information Systems (book) which contains a chapter Developing Your Project Proposal https://link.springer.com/chapter/10.1007/978-1-84800-009-4_5 . While the proposal you will need for this course is smaller than you would be expected to do for a thesis, it follows basically the same form as both proposal types address typical issues arising in undertaking a project.
   4. Extended Quote from Third Class (Monday) 25 Sept 2017 (last year)
      1. from chapter on developing your project proposal: https://link.springer.com/chapter/10.1007/978-1-84800-009-4_5

- the proposal should be short and easy to read (in particular, easy for the marker to find the stuff they are looking for), use one of the two structures for section headings shown in the chapter
  - pp.27 Subject area, Aim, Arguments, Objectives
  - pp.34 Introduction, Reasons, Aim of Project, Objectives
  - for either structure, for the proposal as a whole, you should give it a meaningful title and include your name and email address -- no need for student ID numbers
  - if you compare the two structures, you find that Reasons pretty much Arguments, Aim of Project is pretty much Aim, Introduction is pretty much Subject area, and objectives looks a lot like objectives.
- for a class project, a one page proposal is reasonable and it certainly shouldn’t be more than two pages.
- Subject area, Introduction, Arguments, Reasons: you should pick something that you think would be interesting/useful to you and that you can explain why that is, the more focussed the better. You should give a bit of thought to how one would know whether you were successful or not – that you created something you can build on or something that would need to be redone before further progress was possible.
- Objectives:
  - specific goals for each progress report are needed in order to be able to monitor progress and know when things need to be rethought
  - resources are important. What background do you have in the area chosen?
• resources are important. you are basically looking at a 60 hour project. is it realistic to undertake a measurable amount of progress on your aim in that time frame?
• resources are important. you are probably looking at a 10 page write up at the end. is your project structure well enough organized that it can be reasonably explained in that amount of space?
• resources are important. non-programming projects generally involved reading papers. how long does it take you to read a paper (i generally budget 5 minutes a page for myself, but you may read faster or slower than that – to make plans for your own work, you should know how long things take you).
• resources are important. programming projects are notoriously difficult to estimate the amount of time it will take to do things. a programming project should have an iterative structure, so that if you have to stop early there will still be things done that can be written about and illustrate issues related to the course.
• resources are important. writing, like programming, is notoriously difficult to estimate the amount of time it will take to do things well. budgeting write up time for the end of the project is a recipe for disaster. projects get judged on write ups and presentations, so good project work and poor write ups become poor marks. the solution is to ALWAYS BE WRITING https://www.youtube.com/watch?v=Ya246_Pjikc). as you complete sub goals, include write ups of those sub goals to eventually become the first draft of the final write up. Also, remember the quote from Leslie Lamport: Thinking Above the Code https://www.youtube.com/watch?v=4Yp3j_jk8Q at 3:27, “To think you have to write. If you’re thinking without writing, you only think you’re thinking.” Writing papers in computer science

handing in:
• paper is best as then it doesn’t matter what software you used to create the document. paper is easier to read than computer screens. and paper is much easier to mark up than computer screens.
• the department runs git repositories for class using software called BitBucket. to use this requires getting an account on the system (everyone officially enrolled should have an account – it is the same one you would use to add comments to Confluence). To use the repository, you need to initialize it with an ssh key (see instructions at Atlassian https://www.atlassian.com/software/bitbucket). Note that we are running a local copy and not using the public cloud that Atlassian manages. The local copy is at https://repo.csd.uwo.ca/dashboard. Given that you have a problem with paper, the best format for text is plain text (not Microsoft RTF, Word, WordPerfect, etc.). If you use some sort of odd consumer system that can't manage plain text, PDF is usually acceptable – like most standards, there are many versions of PDF and some don't view well.
• time: it is due at the due date. ideally it would be handed to me in class that day. I will probably end up allowing people to slide their proposal under my door (or submit online) until the end of the day.

5. Quote from Fifth Class Monday 16 Oct 2017 (last year – looking back at what was handed in after above instructions)

One issue that many people missed on their proposals was a discussion of how (of 'Always Be Writing') it will take to do things well. This sort of thing was alluded to in Third Class (Monday) 25 Sept 2017 at the bottom of the page under objectives regarding the importance of resources. Interestingly, if we look at the book Thesis Projects A Guide for Students in Computer Science and Information Systems (book) that was the basis for the third class discussion of proposals, we find that the 'time plan' is not mentioned in proposals, but rather discussed in the subsequent chapter on Developing you Aim https://link.springer.com/chapter/10.1007/978-1-84800-009-4_7 (which is one of the parts of the proposal). Incidentally, the time plan gets a brief mention again in the 2 page Chapter 9 Following The Objectives https://link.springer.com/chapter/10.1007/978-1-84800-009-4_9. And, as was brilliantly said in Third Class (Monday) 25 Sept 2017: specific goals for each progress report are needed in order to be able to monitor progress and know when things need to be rethought.

Also, perhaps of interest in understanding the roles of proposal and progress report is to consider Chapter 3 Actors Involved, their Roles and Relationships https://link.springer.com/chapter/10.1007/978-1-84800-009-4_9. In thesis supervision, it is traditional to exchange information among the actors via conversations and regular meetings. As organizations get larger, such exchanges become written documents. A classroom where many people are working on different tasks is usually organized more like a large organization than a thesis supervision.

Many people have trouble with time estimates. RubyConf 2015 - I Estimate this Talk will be 20 Minutes Long, Give or Take 10 Minutes (talk) and RailsConf 2014 - Unreasonable Estimates and Improbable Goals by Adam Sanderson (talk)

9. General Advice on Handling Reading Assignments in this course (from Second Class (Monday) 18 Sept 2017):

1. as a grad student, you are expected to be able to read material like this and figure out what it is about. Four questions that you should be considering as you look at this material are:
   • What is the main point the author/speaker is trying to get across?
   • Do they make a convincing case? Did you already agree with this or is there something new here? How does this information fit into what else you know about this topic?
   • Was any new vocabulary introduced? Can you find it being used elsewhere, or is it specific to this author/speaker? How does the new vocabulary help organize the presentation?
   • Can you see how this information might be used by you in some future undertaking?

10. Readings for Next Class:

1. on systems: In A World of Systems (video) (9 minute video)
2. on requirements: Requirements Engineering Lecture 1: Overview (video) (10 minute talk)
3. on safety: An integrated approach to safety and security based on systems theory (paper) (5 page paper)
4. on accessibility: Universal Design (talk) (24 minute talk)
5. on sustainability: Towards a definition of sustainability in and for software engineering (paper) (3 page paper)
11. Note: the readings for this course are meant to be an adaptation of the goals of the course outline to the specific interests of the students actually participating in the course. The above readings were assigned before it was known what aspects of the course are of most interest to those taking the course. Give some thought to what interests you would like to see develop. This may or may not be related to the in-depth investigations you plan on doing yourself for the paper in this course.

referenced by: CS9622 Fall 2018 Version of Course

Lectures to an Empty Classroom

First Empty Semester (Expanded First Empty Semester menu):

- First Class -- 9 Feb 2018
- Second Class -- 27 May 2018
- Third Class -- 28 May 2018
- Fourth Class -- 31 May 2018
- Fifth Class -- 19 June 2018
- Sixth Class -- 23 June 2018
- Seventh Lecture -- 6 July 2018
- Eighth Lecture -- 9 July 2018
- Ninth Lecture -- 12 July 2018
- Tenth Lecture -- 16 July 2018
- Eleventh Lecture -- 19 July 2018
- Twelfth Lecture -- 26 July 2018
- Thirteenth Lecture -- 29 July 2018

Second Empty Semester:

- Fourteenth Lecture -- In Progress

Eighth Lecture -- 9 July 2018

FYI:

- Bonnie Nardi, 2017 GAD Distinguished Lecture (talk)
- Heteromation and its (dis)contents: The invisible division of labor between humans and machines (paper)

General:

- RE lecture 5 Artifact Orientation (talk)
- Requirements Engineering - Business Case Analysis (talk)
- RE lecture 6 Problem Vs Solution orientation (talk)
- RE lecture 7 Stakeholders (talk)

Safety:

- A Case Study of Toyota Unintended Acceleration and Software Safety (talk)
- The infeasibility of quantifying the reliability of life-critical real-time software (paper)

Accessibility:

- The Story of Co-Design by thinkpublic (talk)
- Design Anthropology and Co-Design - presented by Kelsey Schwenk at Real Big Things #4 (talk)
- Exploring Co-design with Breastfeeding Mothers (talk) and Exploring Co-design with Breastfeeding Mothers (paper)

Sustainability:

- Computational Agroecology: Sustainable Food Ecosystem Design (talk) and Computational Agroecology: Sustainable Food Ecosystem Design (paper)

Bonnie Nardi, 2017 GAD Distinguished Lecture (talk)

https://www.youtube.com/watch?v=ewuVbMr2zOQ 1:16:09

home: http://www.artifex.org/~bonnie/
Computational Agroecology: Sustainable Food Ecosystem Design (paper)

- Barath Raghavan, Bonnie Nardi, Sarah T. Lovell, Juliet Norton, Bill Tomlinson and Donald J. Patterson
- Pages 423-435
- https://dl.acm.org/citation.cfm?id=2892577

Computational Agroecology: Sustainable Food Ecosystem Design (talk)

https://www.youtube.com/watch?v=KE3XS6oLKhQ 19:32

Design Anthropology and Co-Design - presented by Kelsey Schwenk at Real Big Things #4 (talk)

https://www.youtube.com/watch?v=JQNnGFm7qmc 20:31

Design Serving People: Innovation through Co-creation, Liz Sanders, Northwestern University (talk)

https://www.youtube.com/watch?v=gsFySi8KQoY 58:22

Exploring Co-design with Breastfeeding Mothers (paper)

- Chelsea-Joy Wardle, Mitchell Green, Christine Wanjiru, and Melissa Densmore
- https://dl.acm.org/citation.cfm?doid=3173574.3174056

Exploring Co-design with Breastfeeding Mothers (talk)

https://www.youtube.com/watch?v=DFoVhAvhkgk 21:53

Heteromation and its (dis)contents: The invisible division of labor between humans and machines (paper)

- Ekbia, Hamid; Nardi, Bonnie.
- First Monday, [S.I.], May 2014. ISSN 13960466.

RE lecture 5 Artifact Orientation (talk)

https://www.youtube.com/watch?v=aQ6SpSIRCMU&index=7&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PlNdeQL 8:00

RE lecture 6 Problem Vs Solution orientation (talk)

https://www.youtube.com/watch?v=GfZGAcIzU0&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PlNdeQL&index=9 13:45
RE lecture 7 Stakeholders (talk)

https://www.youtube.com/watch?v=b68Va0Th8-M&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PiNdeQL&index=10  8:41

cf slides: http://birgit.penzenstadler.de/teach/590/2015spring_CECS-590_06_Stakeholders.pptx.pdf

Requirements Engineering - Business Case Analysis (talk)

https://www.youtube.com/watch?v=hAd-YEx4njE&index=8&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PiNdeQL  19:48

slides: https://www.slideshare.net/kamikitty/requirements-engineering-business-case-analysis


The Story of Co-Design by thinkpublic (talk)

https://www.youtube.com/watch?v=HWgJlwTDIRQ  3:47

A Case Study of Toyota Unintended Acceleration and Software Safety (talk)

https://www.youtube.com/watch?v=DKHa7rkvK8  59:59

slides: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=426747 and https://www.slideshare.net/PhilipKoopman/toyota-unintended-acceleration

blog: https://betterembsw.blogspot.com/

The infeasibility of quantifying the reliability of life-critical real-time software (paper)

- R.W. Butler, G.B. Finelli
- IEEE Transactions on Software Engineering (Volume: 19, Issue: 1, Jan 1993)
- Page(s): 3 - 12

Eleventh Lecture -- 19 July 2018

FYI:
- Design Is [Empathic] – A Talk on Empathy, Humanity and Designing AI (talk)
- The Robotic Moment: Who Do We Become When We Talk to Machines? (talk)
- Why Do Virtual Pets Give Us Real Feelings? | Game/Show | PBS Digital Studios (talk)
  - Barayuga, Debra (June 2, 1997). "Electronic Pets Peep Their Last at Isle Schools". Honolulu Star-Bulletin Local News. Archived from the original on 1999-01-29
  - Introducing the Tamagotchi P1 (9:44) https://www.youtube.com/watch?v=cYRABCUoEB0 (1996 original tamagotchi)

General:
- RE lecture 9 System Vision (talk)
- Methods & tools: the rich picture: a tool for reasoning about work context (paper)
- Soft Systems Methodology: The Use of Rich Pictures from Evaluation (talk)

Safety:
• Chapter 6: Engineering and Operating Safer Systems Using STAMP -- Engineering a safer world: systems thinking applied to safety (book)
• Designing and Operating Safety Systems: The Missing Link - SDM alumnus John Helferich (talk)

Accessibility:
• Practical hands-on accessibility testing (talk)

Sustainability:
• Rich pictures: a means to explore the ‘sustainable mind’? (paper)
• DPSIR = A Problem Structuring Method? An exploration from the “Imagine” approach (paper)
• Rich Pictures: Sustainable Development and Stakeholders – The Benefits of Content Analysis (paper)

Chapter 6: Engineering and Operating Safer Systems Using STAMP -- Engineering a safer world: systems thinking applied to safety (book)

• Engineering a safer world: systems thinking applied to safety (book)
• http://sunnyday.mit.edu/safer-world.pdf
• pages 147 – 154

Design Is [Empathic] – A Talk on Empathy, Humanity and Designing AI (talk)

https://www.youtube.com/watch?v=xP1woFRSs_0 52:45

DPSIR = A Problem Structuring Method? An exploration from the “Imagine” approach (paper)

• Simon Bell
• European Journal of Operational Research
• Volume 222, Issue 2, 16 October 2012, Pages 350-360

Methods & tools: the rich picture: a tool for reasoning about work context (paper)

• Andrew Monk and Steve Howard
• interactions, Volume 5 Issue 2, March/April 1998, Pages 21-30
• https://dl.acm.org/citation.cfm?doid=274430.274434

Practical hands-on accessibility testing (talk)

https://www.youtube.com/watch?v=_B1o6DyLCVY 1:33:42

RE lecture 9 System Vision (talk)

https://www.youtube.com/watch?v=zaEfkSXXwY8&list=PLUgFMzvE8lQDeixpbP3s6EyQx8PiNdexQ&index=14 7:28

Rich pictures: a means to explore the ‘sustainable mind’? (paper)

Simon Bell, Tessa Berg, and Stephen Morse
Sustainable Development, Volume 24, Issue 2, March/April 2016, Pages 136-148

Soft Systems Methodology: The Use of Rich Pictures from Evaluation (talk)

https://www.youtube.com/watch?v=q7CTREXiFuk 27:24

The Robotic Moment: Who Do We Become When We Talk to Machines? (talk)

https://www.youtube.com/watch?v=csba3YKD9-A 57:37

Why Do Virtual Pets Give Us Real Feelings? | Game/Show | PBS Digital Studios (talk)

https://www.youtube.com/watch?v=T7rnpdBZvPo 6:23

Expanded First Empty Semester
(first semester portion of Lectures to an Empty Classroom)

First Class -- 9 Feb 2018
- Ruby On Ales 2016: Sharpening The Axe: Self-Teaching For Developers by Aja Hammerly (talk)
- MountainWest RubyConf 2014 - Nerd Party, v 3.1 by Ryan Davis (talk)
- RR 171 - Evaluating Yourself - Ruby Rogues (podcast video)
- YOW! West 2017 Melinda Seckington - Reflect & Refactor (talk)
- Barbara Oakley: “Learning How to Learn” | Talks at Google (talk)
- Accelerating learning from experience: avoiding defects faster (paper)
- Greg Wilson - What We Actually Know About Software Development, and Why We Believe It’s True (talk)
- KOLL-2016-11-25 - Views on external and internal validity in software engineering (Janet Siegmund) (talk)
- NYT E03 (2016) - Controlled Experiments (Siegmund) (talk)

Second Class -- 27 May 2018
FYI: Positive Psychology course at Yale covered on CBC’s morning show The Current:
- an interesting related concept is ‘mindfulness’, cf: What Do We Really Know About Meditation and Health? (UCTV video)

General
- On Non-Functional Requirements (paper)

Safety:
- DevOpsDays Detroit 2017: Day 1 Keynote by Dr. Richard I. Cook (video)
Third Class -- 28 May 2018

FYI:

- George Loewenstein and the NEW New Economics of Information (talk)

General:

- Representing and Using Nonfunctional Requirements: A Process-Oriented Approach (paper)
- Requirements Engineering lecture 1: Overview (video)

Safety:

- Introduction to: Systems Theoretic Accident Model & Processes (STAMP) WEBINAR REPLAY (talk)

Accessibility:

- Evaluating the Accessibility and Usability of Blogging Platforms for Blind Users (paper)

Sustainability:

- Framing sustainability as a property of software quality (paper)
- Characterizing the contribution of quality requirements to software sustainability (paper)

Fourth Class -- 31 May 2018

FYI:

- Evidence shows Starbucks’ anti-bias training may backfire, says expert http://www.cbc.ca/radio/thecurrent/evidence-shows-starbucks-anti-bias-training-may-backfire-says-expert-1.4681343
- Diversity and inclusion research: Javeed Sukhera (talk)
- Why Diversity Programs Fail with Harvard professor Frank Dobbin (talk)
- Interview with Frank Dobbin (Harvard) and Alexandra Kalev (Tel Aviv) on Inequality (talk)

General:

- Requirements Engineering lecture 2: process (talk)
- Al Davis on Entrepreneurship in Software (talk)
- Weaving together requirements and architectures (paper)

Safety:

- Chapter 1: Engineering a safer world: systems thinking applied to safety (book)
- Chapter 2: Engineering a safer world: systems thinking applied to safety (book)

Accessibility:

- Chapter 1: Introduction: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- Chapter 2: Background: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

Sustainability:

- Chapter 1: Introduction and Overview: Software Engineering for Sustainability (thesis)

Fifth Class -- 19 June 2018
FYI:

- Stanford Seminar - Conducting Usable Privacy and Security Studies: It's Complicated (talk)

General:

- Requirements Engineering lecture 3: challenges (talk)
- RE LECTURE 4 Generic Process Model (talk)

Safety:

- Human Factors and Systems Safety Engineering in Healthcare (talk)

Accessibility:

- 08 Sarah Horton: Empathy, Education, and Excellence in Inclusive Design (talk)
- A Day in the Life of a Blind Person (talk)

Sustainability:

- Lynn Bartram - Human-Centred Systems for Sustainable Living (talk)

Sixth Class -- 23 June 2018

FYI:

- GOTO 2016 • Chaos & Intuition Engineering at Netflix • Casey Rosenthal (talk)
- GOTO 2018 • Deprecating Simplicity • Casey Rosenthal (talk)

General:

- ReqEng-ProcessAndPrinciples (talk)
- Requirement Engineering - Frameworks And Standards (talk)
  - https://gephi.org/developers/
- http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_group2.pdf Software Requirements Specification Amazing Lunch Indicator

Safety:

- Safety-Critical Systems - Professor Martyn Thomas CBE (talk)

Accessibility:

- Contextual Enquiry (talk)
- Sarah Horton "Involving People with Disabilities in UX Research" (ID24 2014) (talk)

Sustainability:

- Chapter 17 Domestic Plant Guilds: A Software System for Sustainability: Software Engineering for Sustainability (thesis)
- Plant Guild Composer: An Interactive Online System to Support Back Yard Food Production (promo)

Seventh Lecture -- 6 July 2018

FYI:

- GOTO 2018 • A Strong Belief, Loosely Held: Bringing Empathy to IT • Nirmal Mehta (talk)

General:

- 2. Requirements Definition from MIT 16.842 Fundamentals of Systems Engineering, Fall 2015 (talk)

Safety:

- Chapter 3: Systems Theory and Its Relationship to Safety -- Engineering a safer world: systems thinking applied to safety (book)
- A Philosophical Look at System Dynamics (talk)
Eighth Lecture -- 9 July 2018

FYI:

- Bonnie Nardi, 2017 GAD Distinguished Lecture (talk)
- Heteromation and its (dis)contents: The invisible division of labor between humans and machines (paper)

General:

- RE lecture 5 Artifact Orientation (talk)
- Requirements Engineering - Business Case Analysis (talk)
- RE lecture 6 Problem Vs Solution orientation (talk)
- RE lecture 7 Stakeholders (talk)

Safety:

- A Case Study of Toyota Unintended Acceleration and Software Safety (talk)
- The infeasibility of quantifying the reliability of life-critical real-time software (paper)

Accessibility:

- The Story of Co-Design by thinkpublic (talk)
- Design Anthropology and Co-Design - presented by Kelsey Schwenk at Real Big Things #4 (talk)
- Exploring Co-design with Breastfeeding Mothers (talk) and Exploring Co-design with Breastfeeding Mothers (paper)

Sustainability:

- Computational Agroecology: Sustainable Food Ecosystem Design (talk) and Computational Agroecology: Sustainable Food Ecosystem Design (paper)

Ninth Lecture -- 12 July 2018

FYI:

- Ending Institutional Corruption | Understanding the Human Mind: Insights from Psychology (talk)
- Ending Institutional Corruption | What is Institutional Corruption?: Lessig in the Dock (talk)
- Ending Institutional Corruption | Institutional Corruption in Government and Law (talk)
- Ending Institutional Corruption | Integrity and Trust in Academia and Nonprofits (talk)
- Hacking iCorruption: Results from the Hackathon - May 2, 2015 (talk)

General:

- Requirements Engineering - Elicitation Techniques (talk)
- The Dark Arts of Business Analysis: Requirements Gathering with Mind Maps for a Drupal Project (talk)

Safety:

- Chapter 4: A Systems-Theoretic View of Causality -- Engineering a safer world: systems thinking applied to safety (book)
- 5 min. Intro to: STAMP (Systems Theoretic Accident Model & Processes) (talk)

Accessibility:

- Chapter 4: Research Approach and Methods: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- note parallels between using qualitative methods to address research questions and gathering requirements information from stakeholders
  - Fundamentals of Qualitative Research Methods: What is Qualitative Research (Module 1) (talk)
  - Fundamentals of Qualitative Research Methods: Developing a Qualitative Research Question (Module 2) (talk)
  - Fundamentals of Qualitative Research Methods: Interviews (Module 3) (talk)
  - Fundamentals of Qualitative Research Methods: Focus Groups (Module 4) (talk)
  - Fundamentals of Qualitative Research Methods: Data Analysis (Module 5) (talk)
  - Fundamentals of Qualitative Research Methods: Scientific Rigor (Module 6) (talk)
Sustainability:

- Chapter 7: Who is the advocate?: stakeholders for sustainability: A Software System for Sustainability: Software Engineering for Sustainability (thesis)
- Robin Leichenko: Stakeholder Attitudes on Sustainability and Resiliency (talk)

Tenth Lecture -- 16 July 2018

FYI:

- How to Read a Paper Efficiently (By Prof. Pete Carr) (talk)
- How To Read an Academic Paper (talk)
- How to Read a Research Paper (talk)
- How to read a scientific paper (talk)

General:

- RE lecture 8 Goals (talk)
- Requirements Engineering Goal Modeling (talk)
- Goal-directed concept acquisition in requirements elicitation (paper)
- Collaborative requirements negotiation with EasyWinWin (paper)
- Why Agent-Oriented Requirements Engineering (paper)

Safety:

- Chapter 5: A Friendly Fire Accident -- Engineering a safer world: systems thinking applied to safety (book)

Accessibility:

- Developing a Tool for Testing Compatibility of Websites with ATs (paper) (VATLab)
- Creating Personas with Disabilities (paper) (Personas method)

Sustainability:

- Smart Cities and M3: Rapid Research, Meaningful Metrics and Co-Design (paper)
- Sustainability Indicators Past and Present: What Next? (paper)

Eleventh Lecture -- 19 July 2018

FYI:

- Design Is [Empathic] – A Talk on Empathy, Humanity and Designing AI (talk)
- The Robotic Moment: Who Do We Become When We Talk to Machines? (talk)
- Why Do Virtual Pets Give Us Real Feelings? | Game/Show | PBS Digital Studios (talk)
  - Introducing the Tamagotchi P1 (9:44) https://www.youtube.com/watch?v=cYRABCUoEB0 (1996 original tamagotchi)

General:

- RE lecture 9 System Vision (talk)
- Methods & tools: the rich picture: a tool for reasoning about work context (paper)
- Soft Systems Methodology: The Use of Rich Pictures from Evaluation (talk)

Safety:

- Chapter 6: Engineering and Operating Safer Systems Using STAMP -- Engineering a safer world: systems thinking applied to safety (book)
- Designing and Operating Safety Systems: The Missing Link - SDM alumnus John Helferich (talk)

Accessibility:

- Practical hands-on accessibility testing (talk)

Sustainability:
Twelfth Lecture -- 26 July 2018

FYI:

- 5. Dennis Meadows - Perspectives on the Limits of Growth: It is too late for sustainable development (talk)
- A comparison of The Limits to Growth with 30 years of reality (paper)
- System Dynamics Model: Kaibab Deer Population (talk)
- Systems: Overshoot and Collapse (talk)
- How The Economic Machine Works by Ray Dalio (talk)

General:

- Requirements Engineering Usage Model - learning glass lecture 10 (talk)
- The Use Case Technique - An Overview w/ Karl Wiegers (talk)

Safety:

- Chapter 7: Fundamentals -- Engineering a safer world: systems thinking applied to safety (book)
- SREcon18 Americas - Approaching the Unacceptable Workload Boundary (talk)
- SREcon18 Americas - Whispers in Chaos: Searching for Weak Signals in Incidents (talk)
- On the (In)Security of Mobile Two-Factor Authentication (paper)

Accessibility:

- Secure and Inclusive Authentication with a Talking Mobile One-Time-Password Client (paper)
- see demo of standard version in ProtectID collecting a one-time-password via phone for Out Of Band Two-Factor Authentication (talk)
- on why it doesn't really work Bypassing multi factor authentication - Shahmeer Amir, HackIT-2017 (talk)

Sustainability:

- Chapter 19: Supporting Physicians by RE4S — Evaluating Requirements Engineering for Sustainability in the Medical Domain: Software Engineering for Sustainability (thesis)
- Guidelines for conducting and reporting case study research in software engineering (paper)
- Overcoming Barriers to Medication Adherence for Chronic Diseases (talk)

Thirteenth Lecture -- 29 July 2018

FYI:

  - the GOMS approach to HCI (analyze performance time on tasks as measure of interaction quality) The prospects for psychological science in human-computer interaction (paper)
  - protocol analysis (think-aloud method) used to explore human problem solving On the analysis of human problem solving protocols (paper)
  - Protocol Analysis - Rev’d Edition: Verbal Reports as Data (book)
  - general theory of thinking Soar : an architecture for general intelligence (paper)
  - Unified theories of cognition (book)

General:

- Scaling Requirements Engineering - learning glass lecture 11 (talk)
- Refining Requirements - RE learning glass lecture 12 (talk)
- Reconciling software requirements and architectures with intermediate models (paper)

Safety:

- Chapter 8: STPA: A New Hazard Analysis Technique -- Engineering a safer world: systems thinking applied to safety (book)

Accessibility:

- CHI 2011 SIGCHI Social Impact Award: Clayton Lewis & Alan Newell (talk) – note: Allen at CMU and Alan at Dundee University
  - Global Public Inclusive Infrastructure (GPII) w/captions+description,no music (talk)
  - Design and the Digital Divide: Insights from 40 Years in Computer Support for Older and Disabled People (book)

Sustainability:
Fifth Class -- 19 June 2018

FYI:

- Stanford Seminar - Conducting Usable Privacy and Security Studies: It's Complicated (talk)

General:

- Requirements Engineering lecture 3: challenges (talk)
- RE LECTURE 4 Generic Process Model (talk)

Safety:

- Human Factors and Systems Safety Engineering in Healthcare (talk)

Accessibility:

- 08 Sarah Horton: Empathy, Education, and Excellence in Inclusive Design (talk)
- A Day in the Life of a Blind Person (talk)

Sustainability:

- Lynn Bartram - Human-Centred Systems for Sustainable Living (talk)

2. Requirements Definition from MIT 16.842 Fundamentals of Systems Engineering, Fall 2015 (talk)

https://www.youtube.com/watch?v=J_y2I09rl_I&index=12&list=PLUI4u3cNGP60jMmB53zI6awCKMnABhYx 1:39:51


note, the full set of slides can be found at https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-842-fundamentals-of-systems-engineering-fall-2015/lecture-notes/MIT16_842F15_Ses1SE_Ovr_vw.pdf

the whole MIT OpenCourseWare page for this course can be found at https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-842-fundamentals-of-systems-engineering-fall-2015/


discussion of exercises on slide 21


08 Sarah Horton: Empathy, Education, and Excellence in Inclusive Design (talk)
A Day in the Life of a Blind Person (talk)

https://www.youtube.com/watch?v=elT978hpWMk 35:30

Human Factors and Systems Safety Engineering in Healthcare (talk)

https://www.youtube.com/watch?v=5B3CYLye6vc 42:38

RCA reviews – Root Cause Analysis


The Field Guide to Understanding 'Human Error' (book)

- Sidney Dekker
- 28 November 2014, CRC Press, 248 pages
- https://www.taylorfrancis.com/books/9781317031840

Naming the pain in requirements engineering (paper)

- D Méndez Fernández, Stefan Wagner, Marcos Kalinowski, Michael Felderer, Priscilla Mafra, Antonio Vetró, Tayana Conte, M-T Christiansson, Desmond Greer, Casper Lassenius, Toml Männistö, Maleknaz Nayabi, Markku Olivo, Birgit Penzenstadler, Dietmar Pfahl, Rafael Prikladnicki, Günther Ruhe, André Schekelmann, Sagar Sen, Rodrigo Spinola, Ahmet Tuzcu, Jose Luis de la Vara, Roel Wieringa
- 2017/10/1
- Empirical software engineering 22, 5, 2298-2338
- https://doi.org/10.1007/s10664-016-9451-7

RE LECTURE 4 Generic Process Model (talk)
references NFR (non functional requirements) framework of Martin Glinz, On Non-Functional Requirements (paper)

Requirements Engineering lecture 3: challenges (talk)

references work discussed further in: Naming the pain in requirements engineering (paper)

quote: But in the end, for something this complicated, it's really hard to design products by focus groups. A lot of times, people don't know what they want until you show it to them.


quote: You can't just ask customers what they want and then try to give that to them. By the time you get it built, they'll want something new.

- Interview with Inc Magazine for its "The Entrepreneur of the Decade Award" (1 April 1989), https://en.wikiquote.org/wiki/Steve_Jobs

Stanford Seminar - Conducting Usable Privacy and Security Studies: It's Complicated (talk)

First Class -- 9 Feb 2018

- Ruby On Ales 2016: Sharpening The Axe: Self-Teaching For Developers by Aja Hammerly (talk)
- MountainWest RubyConf 2014 - Nerd Party, v 3.1 by Ryan Davis (talk)
- YOW! West 2017 Melinda Seckington - Reflect & Refactor (talk)
- Barbara Oakley: "Learning How to Learn" | Talks at Google (talk)
- Accelerating learning from experience: avoiding defects faster (paper)
- Greg Wilson - What We Actually Know About Software Development, and Why We Believe It's True (talk)
- NYT E03 (2016) - Controlled Experiments (Siegmund) (talk)
- Barbara Oakley: "Learning How to Learn" | Talks at Google (talk)
- YOW! West 2017 Melinda Seckington - Reflect & Refactor (talk)
- Stanford Seminar - Conducting Usable Privacy and Security Studies: It's Complicated (talk)

Accelerating learning from experience: avoiding defects faster (paper)

- L. Prechelt,
- This paper advocates DLDA (defect logging, defect analysis) as scaled down version of PSP (Personal Software Process -- NOT Playstation Portable), that is easier for people to learn and yet still leads to improvement.
- In addition to providing the rationale for the method, the paper also presents a small controlled experiment done to support the notion that the method is actually useful.
- sidenote:
- Watts Humphrey has written three books on the personal software process, two books on team software process, and five books on managing software projects in the time period from 1989 to 2011. https://en.wikipedia.org/wiki/Watts_Humphrey
- :etonedis
- notes normally takes 15 full work-days of training to learn to use PSP methodology
- most programmers can't just read about it and then use it because they give up before they have practiced it long enough to get benefits from it.
- DLDA is designed to be much less effort to learn and to use.
- you are meant to keep a log of all errors that occur during all phases of the software process (from requirements definition thru to maintenance) -- defect logging
**you are meant to review the logs (once you have a hundred or so errors logged) to look for what are the most expensive defects being generated and figure out a plan to reduce their occurrence.**

**often some of this logging is automated (for example, syntax errors being logged by the build process)**

**sidenote:**

**An existing tool specifically targeting this sort of work are:**


- of course, one could roll one's own by using something like git commits to record stuff a log in a standard format that could then be processed by ad hoc scripts written in a language like Ruby.


- The quote from the article's abstract is: This project wasn't done for the sake of research or academic impact but was actually deployed in Microsoft and has hundreds of users. Currently, CODEMINE is deployed in all major Microsoft product groups: Windows, Windows Phone, Office, Exchange, Lync, SQL, Azure, Bing, and XBox.

- etonedis

- in the experiment which identified defects as failure to pass target test data found an average of 25 defects in 6 hours per participant.

- more experienced programmers got bigger improvements from using DLDA

- most people get benefit in this short experiment from logging rather than analysis of the log data

**Barbara Oakley: "Learning How to Learn" | Talks at Google (talk)**

[https://www.youtube.com/watch?v=vd2dkMlNiw](https://www.youtube.com/watch?v=vd2dkMlNiw)  1:08:41

**MountainWest RubyConf 2014 - Nerd Party, v 3.1 by Ryan Davis (talk)**

[https://www.youtube.com/watch?v=AK-gVWh_vZ8](https://www.youtube.com/watch?v=AK-gVWh_vZ8)  29:19

- **1:43 topic: user groups and study groups**

- **6:24 Seattle.rb version 1.0**

- **plain vanilla `meetup` era**

- **format: presentations**

- **10:05 2.0 Nerd Party! Era**

- **period: weekly**

- **consistent location: cafe’**

- **format: unstructured hacking**

- **13:04 3.0 Nerd Party plus Monthly Social**

- **meet every Tuesday with social on first Tuesday of month**

- **social includes food and presentations**

- **15:53 3.1 Nerd Party plus Monthly Social plus Study Groups Era**

- **weekly study group meetings before nerd party meetings**

- **19:55 Ruby Brigade Recipe**

- **pick your participants**

- **food and drink are always good**

- **no matter what, meet every week**

- **for a successful study group, pick a book**

- **make it challenging**

- **loosely organize a group**

- **meet every week, no matter what**

- **let people fade away, life happens**

- **24:19 surprise lightening talk on nand2tetrix/The Elements of Computing Systems**

**RR 171 - Evaluating Yourself - Ruby Rogues (podcast video)**

[https://www.youtube.com/watch?v=SjUV7uBRWXY](https://www.youtube.com/watch?v=SjUV7uBRWXY)  1:06:43

This video records a discussion among Ruby developers of how do you evaluate yourself and how do you know if you are going in the right direction. They raise a number of issues, but still leave the matter mostly unresolved.

- **6:15 quote: ``if you are sitting still for too long, you are going to wind up being left behind.``**

- **it takes a certain amount of work to stay where you are**

- **10:32 rather than focussing on learning new language detail, one should instead worry about improving on skills like `how to solve problems' or 'how to think like a computer' or `social skills'. these are things that don't change with time unlike language detail.**
• 13:05 what does learning to think like a computer mean? being able to know there is a data structure or algorithm related to what you are doing and being able to track it down and apply it to your task.
• 17:30 avoid trying to quantify things that are hard to quantify -- there isn't a numeric score for how good a programmer you are.
• 22:10 one measure used is 'am I shipping'; keep track of what you are doing on a weekly basis and ask are you accomplishing something each week. -- sometimes learning that something is a bad idea and will never ship is accomplishing something significant
• 25:35 making a 'what I learned today' list, aiming at three or so items a day, recording what is learned so it doesn't get forgotten and have to be relearned.
• 26:20 diversify your mind so you have more things to build on when a problem is encountered.
• 27:37 currently working on 'intrinsic motivation' (why people do things)
• 30:25 tracks where they are on the 'Dreyfus Model of Skill Acquisition' https://en.wikipedia.org/wiki/Dreyfus_model_of_skill_acquisition
• 32:03 troubles with tracking timing themselves. it does get better over time (even if you are bad at it you should still keep practicing as estimating project time is an important skill).
• 34:35 you have to able to accurately evaluate yourself; it helps when working with others and you see how your skills relate to theirs. also you can learn from discussing why something is done a particular way.
• 39:05 maybe what is important is qualitative feedback rather than numbers
• 41:40 keep a list of things you run into that you don't know how to handle well that you can then target learning about
• 43:00 an important thing to strive for is perspective -- seeing how things fit together and what is important
• 47:34 ``feeling bad about my forward motion means I will do more'' dissatisfaction motivates us to do better
• 54:00 how do you evaluate yourself on non-tech skills?
• 54:45 does anyone keep a journal? it encourages reflection.

Ruby On Ales 2016: Sharpening The Axe: Self-Teaching For Developers by Aja Hammerly (talk)

https://www.youtube.com/watch?v=7JD9ZQZMmjo 34:36

YOW! West 2017 Melinda Seckington - Reflect & Refactor (talk)

https://www.youtube.com/watch?v=FMqPBwa34FY 42:55
slides: https://speakerdeck.com/mseckington/reflect-and-refactor

Fourteenth Lecture -- In Progress

FYI:

General:

Safety:

Security:

• Emergent Properties & Security: The Complexity of Security as a Science (paper)

Accessibility:

WCAG related:

• Code in action: Closing the black box of WCAG 2.0, A Latourian reading of Web accessibility (paper)
• WCAG 2.0 - Perceivable (talk)

Sustainability:

Software Energy Consumption:
Long-living Software Systems:

WCAG 2.0 - Perceivable (talk)

https://www.youtube.com/watch?v=t4PY8mnShm4  1:05:34
slides: http://dennisdeacon.com/pres/a11ychi-wcag/#/

Fourth Class -- 31 May 2018

FYI:

- Evidence shows Starbucks' anti-bias training may backfire, says expert http://www.cbc.ca/radio/thecurrent/evidence-shows-starbucks-anti-bias-training-may-backfire-says-expert-1.4681343
- Diversity and inclusion research: Javeed Sukhera (talk)
- Why Diversity Programs Fail with Harvard professor Frank Dobbin (talk)
- Interview with Frank Dobbin (Harvard) and Alexandra Kalev (Tel Aviv) on Inequality (talk)

General:

- Requirements Engineering lecture 2: process (talk)
- Al Davis on Entrepreneurship in Software (talk)
- Weaving together requirements and architectures (paper)

Safety:

- Chapter 1: Engineering a safer world: systems thinking applied to safety (book)
- Chapter 2: Engineering a safer world: systems thinking applied to safety (book)

Accessibility:

- Chapter 1: Introduction: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- Chapter 2: Background: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

Sustainability:

- Chapter 1: Introduction and Overview: Software Engineering for Sustainability (thesis)

Al Davis on Entrepreneurship in Software (talk)

https://www.youtube.com/watch?v=-DtSCj4qiWs  8:17

Chapter 1: Engineering a safer world: systems thinking applied to safety (book)

- Engineering a safer world: systems thinking applied to safety (book)

Chapter 1: Introduction: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

- Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- pages 3 – 15

Chapter 1: Introduction and Overview: Software Engineering for Sustainability (thesis)

- Software Engineering for Sustainability (thesis)
Chapter 2: Background: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

- Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- pages 16 – 28

Chapter 2: Engineering a safer world: systems thinking applied to safety (book)

- Engineering a safer world: systems thinking applied to safety (book)
- pages 7 – 50.

Diversity and inclusion research: Javeed Sukhera (talk)

https://www.youtube.com/watch?v=nxgHKbFhkKc 13:39

Requirements Engineering lecture 2: process (talk)

https://www.youtube.com/watch?v=GXPfl1bQxUU&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PlNdeQL&index=3 14:04

Weaving together requirements and architectures (paper)

- B. Nuseibeh
- Computer (Volume: 34, Issue: 3, Mar 2001 ), 115 - 119

home page: http://www.open.ac.uk/people/ban25

Why Diversity Programs Fail with Harvard professor Frank Dobbin (talk)

https://www.youtube.com/watch?v=FE3DkmC8jqo 3:36

Interview with Frank Dobbin (Harvard) and Alexandra Kalev (Tel Aviv) on Inequality (talk)

https://www.youtube.com/watch?v=EZK2wgnh_R0 14:21

Ninth Lecture -- 12 July 2018

FYI:

- Ending Institutional Corruption | Understanding the Human Mind: Insights from Psychology (talk)
- Ending Institutional Corruption | What is Institutional Corruption?: Lessig in the Dock (talk)
- Ending Institutional Corruption | Institutional Corruption in Government and Law (talk)
- Ending Institutional Corruption | Integrity and Trust in Academia and Nonprofits (talk)
- Hacking iCorruption: Results from the Hackathon - May 2, 2015 (talk)

General:
• Requirements Engineering - Elicitation Techniques (talk)
• The Dark Arts of Business Analysis: Requirements Gathering with Mind Maps for a Drupal Project (talk)

Safety:
• Chapter 4: A Systems-Theoretic View of Causality -- Engineering a safer world: systems thinking applied to safety (book)
• 5 min. Intro to: STAMP (Systems Theoretic Accident Model & Processes) (talk)

Accessibility:
• Chapter 4: Research Approach and Methods: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
• note parallels between using qualitative methods to address research questions and gathering requirements information from stakeholders
  • Fundamentals of Qualitative Research Methods: What is Qualitative Research (Module 1) (talk)
  • Fundamentals of Qualitative Research Methods: Developing a Qualitative Research Question (Module 2) (talk)
  • Fundamentals of Qualitative Research Methods: Interviews (Module 3) (talk)
  • Fundamentals of Qualitative Research Methods: Focus Groups (Module 4) (talk)
  • Fundamentals of Qualitative Research Methods: Data Analysis (Module 5) (talk)
  • Fundamentals of Qualitative Research Methods: Scientific Rigor (Module 6) (talk)

Sustainability:
• Chapter 7: Who is the advocate?: stakeholders for sustainability: A Software System for Sustainability: Software Engineering for Sustainability (thesis)
• Robin Leichenko: Stakeholder Attitudes on Sustainability and Resiliency (talk)

5 min. Intro to: STAMP (Systems Theoretic Accident Model & Processes) (talk)

https://www.youtube.com/watch?v=fy6DDUVw6cl 4:53

A dynamic and cyclical model of bounded ethicality (paper)

• Dolly Chugh, Mary C. Kern
• Research in Organizational Behavior, Volume 36, 2016, Pages 85-100
• https://www.sciencedirect.com/science/article/pii/S0191308516300041

Chapter 4: A Systems-Theoretic View of Causality -- Engineering a safer world: systems thinking applied to safety (book)

• Engineering a safer world: systems thinking applied to safety (book)
• http://sunnyday.mit.edu/safer-world.pdf
• pages 65 – 87

Chapter 4: Research Approach and Methods: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

• Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
• pages 63 – 96

Chapter 7: Who is the advocate?: stakeholders for sustainability: A Software System for Sustainability: Software Engineering for Sustainability (thesis)

• Software Engineering for Sustainability (thesis)
• pages 97 – 106
• see published version: Who is the advocate?: stakeholders for sustainability (paper)

Cheating more for less: Upward social comparisons motivate the poorly compensated to cheat (paper)

• Leslie K. John, George Loewenstein, Scott I. Rick
• Organizational Behavior and Human Decision Processes
• Volume 123, Issue 2, March 2014, Pages 101-109
Ending Institutional Corruption | Institutional Corruption in Government and Law (talk)

https://www.youtube.com/watch?v=fN_SsRzuNOk  1:15:43

Ending Institutional Corruption | Integrity and Trust in Academia and Nonprofits (talk)

https://www.youtube.com/watch?v=m4uAHS53Qak  1:14:55

for more on Garry Gray's comments see Organizational SelfCensorship: Corporate Sponsorship, Nonprofit Funding, and the Educational Experience (paper)

his more recent work can be seen in Academic Voice in Scholarly Writing (paper)

Academic Voice in Scholarly Writing (paper)

- Garry Gray,
  The Qualitative Report, Volume 22, Number 1 (2017), 20 pages.
  https://nsuworks.nova.edu/tqr/vol22/iss1/10/

Organizational SelfCensorship: Corporate Sponsorship, Nonprofit Funding, and the Educational Experience (paper)

- GARRY C. GRAY, VICTORIA BISHOP KENDZIA
- Canadian Review of Sociology, Volume 46, Issue 2, May 2009, Pages 161-177

Ending Institutional Corruption | Understanding the Human Mind: Insights from Psychology (talk)

https://www.youtube.com/watch?v=aOgPIVCw7D4  1:16:57

Chugh's presentation further discussed in A dynamic and cyclical model of bounded ethicality (paper)

Loewenstein's presentation further discussed in Cheating more for less: Upward social comparisons motivate the poorly compensated to cheat (paper)

Ending Institutional Corruption | What is Institutional Corruption?: Lessig in the Dock (talk)

https://www.youtube.com/watch?v=g6Up1p7Ltkc  55:04

references: Testing Theories of American Politics: Elites, Interest Groups, and Average Citizens (paper)

Testing Theories of American Politics: Elites, Interest Groups, and Average Citizens (paper)

- Martin Gilens and Benjamin I. Page
  Perspective on Politics, Volume 12, Issue 3, September 2014, pp. 564-581
  https://doi.org/10.1017/S1537592714001595

Fundamentals of Qualitative Research Methods: Data Analysis (Module 5) (talk)

https://www.youtube.com/watch?v=opp5tH4uD-w  17:11

Fundamentals of Qualitative Research Methods: Developing a Qualitative Research Question (Module 2) (talk)
Fundamentals of Qualitative Research Methods: Focus Groups (Module 4) (talk)

Fundamentals of Qualitative Research Methods: Interviews (Module 3) (talk)

Fundamentals of Qualitative Research Methods: Scientific Rigor (Module 6) (talk)

Fundamentals of Qualitative Research Methods: What is Qualitative Research (Module 1) (talk)

Hacking iCorruption: Results from the Hackathon - May 2, 2015 (talk)

general discussion of hackathons: https://en.wikipedia.org/wiki/Hackathon


cf live blog of event https://medium.com/3-to-read/hacking-icorruption-f279e753882b

Unearth browser extension at chrome store https://chrome.google.com/webstore/detail/unearth/nlgekenmjjflbaohkglefebfdinbmi?hl=en

Learning is... Hard | Russell Sarder feat. Dolly Chugh | Series 185 (talk)

Requirements Engineering - Elicitation Techniques (talk)

Robin Leichenko: Stakeholder Attitudes on Sustainability and Resiliency (talk)
The Dark Arts of Business Analysis: Requirements Gathering with Mind Maps for a Drupal Project (talk)

Second Class -- 27 May 2018

FYI: Positive Psychology course at Yale covered on CBC's morning show The Current:

- an interesting related concept is "mindfulness", cf: What Do We Really Know About Meditation and Health? (UCTV video)

General

- [http://www.coursera.org/learn/the-science-of-well-being](http://www.coursera.org/learn/the-science-of-well-being)

Safety:

- DevOpsDays Detroit 2017: Day 1 Keynote by Dr. Richard I. Cook (video)

Accessibility:

- Accessibility for Everyone - An Evening with Laura Kalbag (video)
- [https://en.wikipedia.org/wiki/Accessibility](https://en.wikipedia.org/wiki/Accessibility)

Sustainability:

- Patricia Lago: Sustainable Software for a Digital Society (video)

Accessibility for Everyone - An Evening with Laura Kalbag (video)

[https://www.youtube.com/watch?v=zQSV1pKSr3s](https://www.youtube.com/watch?v=zQSV1pKSr3s) 1:14:08


interview/review: [https://theuxreview.co.uk/accessibility-for-everyone-interview-laura-kalbag/](https://theuxreview.co.uk/accessibility-for-everyone-interview-laura-kalbag/)
blog: The first 8 reasons to read Accessibility for Everyone by Laura Kalbag https://hk.saowen.com/a/5a7e8c014a839183d4c829dc244c09c6f9758271d4d6f198339684ae64e38

Ethical Design Manifesto: https://2017.ind.ie/ethical-design/

**DevOpsDays Detroit 2017: Day 1 Keynote by Dr. Richard I. Cook (video)**

https://www.youtube.com/watch?v=4jRqtciRQds 43:24


sidenote: see also: “Going solid”: a model of system dynamics and consequences for patient safety; R Cook, J Rasmussen - BMJ Quality & Safety, 2005, 5 pages. http://qualitysafety.bmj.com/content/14/2/130

notes:

- quote from David Woods, 2017: As the complexity of a system increases, the accuracy of any agent's model of that system decreases.  
- quote: connor's corollary: after accident inquiries model the organizational factors that produced the event  
- quote: continuous deployment requires continuous scrutiny  
- https://www.snaucatchers.com/  

**On Non-Functional Requirements (paper)**

- Martin Glinz  
- Requirements Engineering Conference, 2007. RE ’07. 15th IEEE International  
- 6 pages.  

notes:

- functional requirements assume a program can be described as a mathematical function.  
- since programs run on computers, sometimes required execution resources, such as time and memory, are thought of as part of functional requirements.  
- various definitions from the literature for the term: non-functional requirements  
- Section 4.2 proposes that requirements be categorized into: functional requirements, performance requirements, specific quality requirements, and constraints on the system not covered by the other three categories. Non-functional requirements are then just the three categories that aren't functional.  
- Table 3 gives classification rules for determining what category a requirement falls into.

**Patricia Lago: Sustainable Software for a Digital Society (video)**

https://www.youtube.com/watch?v=3yDbwh8-ZJ0 28:23

slides for previous year's version of this talk: https://www.slideshare.net/patricia_lago/software-and-sustainability and accompanying text https://research.vu.nl/en/publications/software-and-sustainability

**What Do We Really Know About Meditation and Health? (UCTV video)**

https://www.youtube.com/watch?v=L3kI4GXHet8 1:27:17

**Seventh Lecture -- 6 July 2018**
FYI:
- GOTO 2018 • A Strong Belief, Loosely Held: Bringing Empathy to IT • Nirmal Mehta (talk)

General:
- 2. Requirements Definition from MIT 16.842 Fundamentals of Systems Engineering, Fall 2015 (talk)

Safety:
- Chapter 3: Systems Theory and Its Relationship to Safety -- Engineering a safer world: systems thinking applied to safety (book)
- A Philosophical Look at System Dynamics (talk)

Accessibility:
- Chapter 3: Previous and Related Research: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- Open Source & Feelings 2015 - Designing with Empathy by Greg Tarnoff (talk)

Sustainability:
- Goal modeling for sustainability: The case of time (paper)
- What is Permaculture? Bill Mollison, David Holmgren (talk)

Chapter 3: Previous and Related Research: Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- pages 29 – 62

Chapter 3: Systems Theory and Its Relationship to Safety -- Engineering a safer world: systems thinking applied to safety (book)
- Engineering a safer world: systems thinking applied to safety (book)
- pages 51 – 60

Goal modeling for sustainability: The case of time (paper)
- Gunter Mussbacher : Douglas Nuttall
- 2014 IEEE 4th International Model-Driven Requirements Engineering Workshop (MoDRE), 10 pages.

GOTO 2018 • A Strong Belief, Loosely Held: Bringing Empathy to IT • Nirmal Mehta (talk)

https://www.youtube.com/watch?v=Lyza-qFltus 45:13

twitter: https://twitter.com/normalfaults?ref_src=twsrc%5Egoogle%7Ctwcamp%5Eserp%7Ctwgr%5Eauthor


references 1.5 year program at George Mason University (Virginia) for MS in Government Contracts https://business.gwu.edu/academics/programs/specialized-masters/ms-government-contracts/faqs
game theory ideas expanded from: Breaking Bad Equilibrium - John Willis (talk)

organizational scar tissue from The Future of Cloud Innovation, featuring Adrian Cockcroft https://www.slideshare.net/Indicee/cloud-trends-2015-pdf – for more on Adrian Cockcroft see GOTO 2017 • Cloud Trends • Adrian Cockcroft (talk)

references active listening https://en.wikipedia.org/wiki/Active_listening


references Docker https://en.wikipedia.org/wiki/Docker_(software)

Breaking Bad Equilibrium - John Willis (talk)

https://www.youtube.com/watch?v=W3h3cyFteXs&feature=youtu.be&list=PLotLY1RC8HosrR60aQyafck5qWn2cTeQq 31:25

slides: https://www.slideshare.net/SeniorStoryteller/breaking-bad-equilibrium-john-willis

presentations list: https://github.com/botchagalupe/my-presentations


references Simon Wardley ( GOTO 2016 • Situation Normal, Everything Must Change • Simon Wardley (talk) ), see also https://medium.com/@johnwillis/the-calculus-of-decision-making-9721d66e5620

references: there is no talent shortage -- velocity NYC 2013 -- andrew clay shafer (talk)


GOTO 2016 • Situation Normal, Everything Must Change • Simon Wardley (talk)

https://www.youtube.com/watch?v=fYH-vLWHhyY 44:01


home site: https://medium.com/wardleymaps

references "GCHQ’s internal Boiling Frogs research paper on software development and organisational change in the face of disruption " https://github.com/gchq/BoilingFrogs

there is no talent shortage -- velocity NYC 2013 -- andrew clay shafer (talk)

https://www.youtube.com/watch?v=P_sWGl7MzhU 38:05

slides: https://www.slideshare.net/littleidea/there-is-no-talent-shortage-velocity-2013

references Patty McCord, cf: Patty McCord: A Culture of Innovation (talk) and Patty McCord: "Powerful: Teams, Leaders and the Culture of Freedom [...] | Talks at Google (talk)

references:


Patty McCord: "Powerful: Teams, Leaders and the Culture of Freedom [...] | Talks at Google (talk)
https://www.youtube.com/watch?v=thzDy5A-KIE  40:49

Patty McCord: A Culture of Innovation (talk)

https://www.youtube.com/watch?v=o3e1InixKBM  11:46

GOTO 2017 • Cloud Trends • Adrian Cockcroft (talk)

https://www.youtube.com/watch?v=EDZBYbEwhm8  51:41
references BMW's CARASSO (cars as sensors): https://aws.amazon.com/solutions/case-studies/bmw/

Open Source & Feelings 2015 - Designing with Empathy by Greg Tarnoff (talk)

https://www.youtube.com/watch?v=KtcM4l5qd4A  32:09

What is Permaculture? Bill Mollison, David Holmgren (talk)

https://www.youtube.com/watch?v=QBLKuYDh5S8  4:16

Sixth Class -- 23 June 2018

FYI:

- GOTO 2016 • Chaos & Intuition Engineering at Netflix • Casey Rosenthal (talk)
- GOTO 2018 • Deprecating Simplicity • Casey Rosenthal (talk)

General:

- ReqEng-ProcessAndPrinciples (talk)
- Requirement Engineering - Frameworks And Standards (talk)
  - https://gephi.org/developers/
  - https://eu-smartcities.eu/
- http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_group2.pdf  Software Requirements Specification
  Amazing Lunch Indicator

Safety:

- Safety-Critical Systems - Professor Martyn Thomas CBE (talk)

Accessibility:

- Contextual Enquiry (talk)
• Sarah Horton “Involving People with Disabilities in UX Research” (ID24 2014) (talk)

Sustainability:
• Chapter 17 Domestic Plant Guilds: A Software System for Sustainability: Software Engineering for Sustainability (thesis)
• Plant Guild Composer: An Interactive Online System to Support Back Yard Food Production (promo)

Chapter 17 Domestic Plant Guilds: A Software System for Sustainability: Software Engineering for Sustainability (thesis)

• Software Engineering for Sustainability (thesis)
• pages 203 – 212

Contextual Enquiry (talk)

https://www.youtube.com/watch?v=mOWeNnSY5M0 3:20

GOTO 2016 • Chaos & Intuition Engineering at Netflix • Casey Rosenthal (talk)

https://www.youtube.com/watch?v=Q4nniyAarbs 49:04
references: http://principlesofchaos.org/
references: https://github.com/Netflix/vizceral

GOTO 2018 • Deprecating Simplicity • Casey Rosenthal (talk)

https://www.youtube.com/watch?v=DtRy79jIsS8 29:09

Plant Guild Composer: An Interactive Online System to Support Back Yard Food Production (promo)

https://www.youtube.com/watch?v=1i97BktwYp0 0:31

ReqEng-ProcessAndPrinciples (talk)

https://www.youtube.com/watch?v=ZzO3oUA7pNw&index=5&list=PLUgFMzuE8lQDeixpP3s6EyQx8PiNdeQL 15:16

Requirement Engineering - Frameworks And Standards (talk)

https://www.youtube.com/watch?v=fYAHXktQ2uI&index=6&list=PLUgFMzuE8lQDeixpP3s6EyQx8PiNdeQL 13:20
references Volere requirements framework: http://www.volere.co.uk/faq.htm


Sarah Horton "Involving People with Disabilities in UX Research" (ID24 2014) (talk)

https://www.youtube.com/watch?v=jxN66EjNUy0 59:40

gap in audio recording between 33 minutes in and 38 minutes in (the auto-generated English transcript gaps from 33:27 to 37:03). and again between 42 minutes and 45 minutes (transcript gaps from 41:50 to 44:37). In general, the audio quality is low, and given the topic, it is worth considering how that impacts the viewing experience in a world where all audio was like this.


references signature guides for the blind: http://www.braillebookstore.com/Metal-Signature-Guide.1

Tenth Lecture -- 16 July 2018

FYI:

- How to Read a Paper Efficiently (By Prof. Pete Carr) (talk)
- How To Read an Academic Paper (talk)
- How to Read a Research Paper (talk)
- How to read a scientific paper (talk)

General:

- RE lecture 8 Goals (talk)
- Requirements Engineering Goal Modeling (talk)
- Goal-directed concept acquisition in requirements elicitation (paper)
- Collaborative requirements negotiation with EasyWinWin (paper)
- Why Agent-Oriented Requirements Engineering (paper)

Safety:

- Chapter 5: A Friendly Fire Accident -- Engineering a safer world: systems thinking applied to safety (book)

Accessibility:

- Developing a Tool for Testing Compatibility of Websites with ATs (paper) (VATLab)
- Creating Personas with Disabilities (paper) (Personas method)

Sustainability:

- Smart Cities and M3: Rapid Research, Meaningful Metrics and Co-Design (paper)
- Sustainability Indicators Past and Present: What Next? (paper)

Chapter 5: A Friendly Fire Accident -- Engineering a safer world: systems thinking applied to safety (book)

- Engineering a safer world: systems thinking applied to safety (book)
- pages 89 – 142

Collaborative requirements negotiation with EasyWinWin (paper)

- P. Gruenbacher

Creating Personas with Disabilities (paper)

- Schulz T., Skeide Fuglerud K.
- https://link.springer.com/chapter/10.1007/978-3-642-31534-3_22

see discussion in Section 4.4.5 of Inclusive design of ICT: The challenge of diversity (U Oslo thesis)

as noted on page 96, three of the projects included in this thesis (Inclusive design of ICT: The challenge of diversity (U Oslo thesis) ) were evaluated using the Personas method.

Developing a Tool for Testing Compatibility of Websites with ATs (paper)

- Stein Erik Skotkjerra, Kristin Skeide Fuglerud, and Till Halbach
- Journal on Technology and Persons with Disabilities, pp 77 -- 88
- Santiago, J. (Eds): Annual International Technology and Persons with Disabilities Conference
- http://scholarworks.csun.edu/bitstream/handle/10211.3/151188/JTPD-2015-p077.pdf?sequence=1


- Aleksander Bai, Heidi Camilla Mork, Trenton Schulz, Kristin Skeide Fuglerud
- Pages 506 - 516
- http://ebooks.iospress.nl/publication/44532

Goal-directed concept acquisition in requirements elicitation (paper)

- Anne Dardenne, Stephen Fickas, and Axel van Lamsweerde
- IWSSD '91 Proceedings of the 6th international workshop on Software specification and design
- Pages 14-21
- https://dl.acm.org/citation.cfm?id=952790

How to Read a Book by Mortimer Adler  Animated Book Summary (talk)

https://www.youtube.com/watch?v=E4p8Jkl9lV8  5:58

How To Read an Academic Paper (talk)
https://www.youtube.com/watch?v=SKxm2HF_-k0  3:13

How to Read a Paper Efficiently (By Prof. Pete Carr) (talk)

https://www.youtube.com/watch?v=leaD0ZaUJ3Y  7:39

How to Read a Research Paper (talk)

https://www.youtube.com/watch?v=SHTOi0KtZnU  8:43

How to read a scientific paper (talk)

https://www.youtube.com/watch?v=5Eg_Gzz3hXY  7:22

How to Use Mind Mapping for Studying and Research (talk)

https://www.youtube.com/watch?v=_3r6ZbE3ci0  2:48

RE lecture 8 Goals (talk)

https://www.youtube.com/watch?v=nW3hWzkoijk&list=PLUgFMzuE8lODeixpbP3s6EyQx8PlNdeQL&index=12  10:32

Requirements Engineering Goal Modeling (talk)

https://www.youtube.com/watch?v=A9KAwiRI8lg&list=PLUgFMzuE8lODeixpbP3s6EyQx8PlNdeQL&index=13  24:01
cf slides: https://eee.uci.edu/14w/37030/home/Inf113_06-GoalsAndConstraints.pptx.pdf

Smart Cities and M3: Rapid Research, Meaningful Metrics and Co-Design (paper)

- Simon Bell, Francesca Benatti, Neil R. Edwards, Robin Laney, David R. Morse, Lara Piccolo, and Oliver Zanetti
- Systemic Practice and Action Research
- February 2018, Volume 31, Issue 1, pp 27–53
- https://link.springer.com/article/10.1007/s11213-017-9415-x

SQ4R and Cornell Notes (talk)
Sustainability Indicators Past and Present: What Next? (paper)

- Simon Bell and Stephen Morse
- Sustainability 2018, 10(5), 1688;
- https://doi.org/10.3390/su10051688

Why Agent-Oriented Requirements Engineering (paper)

- Eric S. K. Yu

Third Class -- 28 May 2018

FYI:

- George Loewenstein and the NEW New Economics of Information (talk)

General:

- Representing and Using Nonfunctional Requirements: A Process-Oriented Approach (paper)
- Requirements Engineering lecture 1: Overview (video)

Safety:

- Introduction to: Systems Theoretic Accident Model & Processes (STAMP) WEBINAR REPLAY (talk)

Accessibility:

- Evaluating the Accessibility and Usability of Blogging Platforms for Blind Users (paper)

Sustainability:

- Framing sustainability as a property of software quality (paper)
- Characterizing the contribution of quality requirements to software sustainability (paper)

Characterizing the contribution of quality requirements to software sustainability (paper)

- Nelly Condori-Fernandez and Patricia Lago

Evaluating the Accessibility and Usability of Blogging Platforms for Blind Users (paper)

- https://doi.org/10.1007/978-1-4471-2867-0_5

George Loewenstein and the NEW New Economics of Information (talk)

https://www.youtube.com/watch?v=o55GZt_sII4  1:17:39

talk starts at 9:20

Requirements Engineering lecture 1: Overview (video)
Thirteenth Lecture -- 29 July 2018

FYI:
  - The GOMS approach to HCI (analyze performance time on tasks as measure of interaction quality)
  - Protocol analysis (think-aloud method) used to explore human problem solving:
    - On the analysis of human problem solving protocols (paper)
    - Protocol Analysis - Rev'd Edition: Verbal Reports as Data (book)
  - General theory of thinking
  - Soar: an architecture for general intelligence (paper)
  - Unified theories of cognition (book)

General:
- Scaling Requirements Engineering - learning glass lecture 11 (talk)
- Refining Requirements - RE learning glass lecture 12 (talk)
- Reconciling software requirements and architectures with intermediate models (paper)

Safety:
- Chapter 8: STPA: A New Hazard Analysis Technique -- Engineering a safer world: systems thinking applied to safety (book)

Accessibility:
- CHI 2011 SIGCHI Social Impact Award: Clayton Lewis & Alan Newell (talk) – note: Allen at CMU and Alan at Dundee University
  - Global Public Inclusive Infrastructure (GPII) w/captions+description.no music (talk)
  - Design and the Digital Divide: Insights from 40 Years in Computer Support for Older and Disabled People (book)

Sustainability:
- "Google: Searching for a Circular Transition" by Kate Brandt (talk)
  - Re-thinking Progress: The Circular Economy (talk)
  - Kate Brandt, Google - WCS Talks 2016 (talk)
  - https://environment.google/projects/airview/
  - https://www.google.com/get/sunroof/#p=0
  - https://waymo.com/

"Google: Searching for a Circular Transition" by Kate Brandt (talk)

https://www.youtube.com/watch?v=amZhVvOMX1g 1:00:10


https://www.youtube.com/watch?v=_sD42h9d1pk 1:14:00

Chapter 8: STPA: A New Hazard Analysis Technique -- Engineering a safer world: systems thinking applied to safety (book)

- Engineering a safer world: systems thinking applied to safety (book)
- pages 179 – 212

CHI 2011 SIGCHI Social Impact Award: Clayton Lewis & Alan Newell (talk)
Design and the Digital Divide: Insights from 40 Years in Computer Support for Older and Disabled People (book)

- Alan F. Newell
- Synthesis Lectures on Assistive, Rehabilitative, and Health-Preserving Technologies
- June 2011, 195 pages
- https://doi.org/10.2200/S00369ED1V01Y201106ARH001

Global Public Inclusive Infrastructure (GPII) w/captions+description,no music (talk)

In Pursuit of Mind: The Research of Allen Newell (paper)

- John E. Laird, Paul S. Rosenbloom

Kate Brandt, Google - WCS Talks 2016 (talk)

On the analysis of human problem solving protocols (paper)

- Allen Newell
- Carnegie Mellon University, 1966, 68 pages


- K. Anders Ericsson and Herbert A. Simon
- A Bradford Book; revised edition edition (April 13, 1993), 496 pages
- UWU Library Catalog: https://uwo.summon.serialssolutions.com/#!/search?bookMark=ePnHCXMw42LgTQSzoc4rAe_hSmGHHjZAbMGdDT6gZOEHTplYQobeTIDVE7CSr-RkEAwoyi_JB_pe1RF6HAcPA0IUUSmwlJRxw1x9tAFlv3x/DfEGN-CRQnPYoAABNyeKVBgBSJSZX

Reconciling software requirements and architectures with intermediate models (paper)

- Paul Grünbacher, Alexander Egyed, and Nenad Medvidovic
Refining Requirements - RE learning glass lecture 12 (talk)

https://www.youtube.com/watch?v=yS8GtR8w2X8&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PNdeQL&index=17 5:17

Re-thinking Progress: The Circular Economy (talk)

https://www.youtube.com/watch?v=zCRKvDyyHml 3:48

Scaling Requirements Engineering - learning glass lecture 11 (talk)

https://www.youtube.com/watch?v=NQ6-JaU3Ow8&index=16&list=PLUgFMzuE8lQDeixpbP3s6EyQx8PNdeQL 12:57
cf slides: http://birgit.penzenstadler.de/teach/590/2015spring_CECS-590_HowToScaleRE.pptx.pdf

Soar : an architecture for general intelligence (paper)

- John Laird, Allen Newell, and Paul S. Rosenbloom
- https://figshare.com/articles/Soar_an_architecture_for_general_intelligence/6609584

The prospects for psychological science in human-computer interaction (paper)

- Allen Newell and Stuart K. Card
- Human-Computer Interaction, Volume 1 Issue 3, September 1985, Pages 209-242
- https://dl.acm.org/citation.cfm?id=1453662

Unified theories of cognition (book)

- Newell, Allen
- Harvard University Press, 1990, 549 pages
- UWO Library Catalog: https://uw summon serialsolutions.com/#/search?bookMark=ePrnHCXMW42LgTQSzc4rAe_hSmEG9oTMzE0sOVs0gRUJb2kDHTqHSdI4aHs4-mESnNnXEvyaALBGnNAVpcCeeAmfsJEokJ-mAF1CA9pcV5AesMGXoXEOcPXSBIUA8dHwJPsY2OqxANpgREaAFoCKpg

Twelfth Lecture -- 26 July 2018

FYI:

- 5. Dennis Meadows - Perspectives on the Limits of Growth: It is too late for sustainable development (talk)
- A comparison of The Limits to Growth with 30 years of reality (paper)
- System Dynamics Model: Kaibab Deer Population (talk)
- Systems: Overshoot and Collapse (talk)
- How The Economic Machine Works by Ray Dalio (talk)

General:

- Requirements Engineering Usage Model - learning glass lecture 10 (talk)
- The Use Case Technique - An Overview w/ Karl Wiegers (talk)
Safety:

- Chapter 7: Fundamentals -- Engineering a safer world: systems thinking applied to safety (book)
- SREcon18 Americas - Approaching the Unacceptable Workload Boundary (talk)
- SREcon18 Americas - Whispers in Chaos: Searching for Weak Signals in Incidents (talk)
- On the (In)Security of Mobile Two-Factor Authentication (paper)

Accessibility:

- Secure and Inclusive Authentication with a Talking Mobile One-Time-Password Client (paper)
- see demo of standard version in ProtectID collecting a one-time-password via phone for Out Of Band Two-Factor Authentication (talk)
- on why it doesn't really work Bypassing multi factor authentication - Shahmeer Amir, HackIT-2017 (talk)

Sustainability:

- Chapter 19: Supporting Physicians by RE4S — Evaluating Requirements Engineering for Sustainability in the Medical Domain: Software Engineering for Sustainability (thesis)
- Guidelines for conducting and reporting case study research in software engineering (paper)
- Overcoming Barriers to Medication Adherence for Chronic Diseases (talk)

5. Dennis Meadows - Perspectives on the Limits of Growth: It is too late for sustainable development (talk)

https://www.youtube.com/watch?v=f2oyU0RusiA  48:23

A comparison of The Limits to Growth with 30 years of reality (paper)

- Graham M. Turner
- Global Environmental Change
- Volume 18, Issue 3, August 2008, Pages 397-411

Bypassing multi factor authentication - Shahmeer Amir, HackIT-2017 (talk)

https://www.youtube.com/watch?v=4Y_NQbNQLg8  24:37

slides: https://www.slideshare.net/HackIT-ukraine/bypassing-two-factor-authentication-shahmeer-amir
blog entry: https://shahmeeramir.com/4-methods-to-bypass-two-factor-authentication-2b0075d9eb5f
referencing: How to Bypass Two-Factor Authentication (2FA) and What the Future Holds (paper)
referencing: How I bypassed 2-Factor-Authentication on Google, Facebook, Yahoo, LinkedIn, and many others (blog)
referencing: Here’s How an Attacker Can Bypass Your Two-Factor Authentication (blog)
Here’s How an Attacker Can Bypass Your Two-Factor Authentication (blog)

- Chris Hoffman on March 18th, 2015
- https://www.howtogeek.com/212219/heres-how-an-attacker-can-bypass-your-two-factor-authentication/

How I bypassed 2-Factor-Authentication on Google, Facebook, Yahoo, LinkedIn, and many others (blog)

- Shubham Shah
- 3 May 2014
- https://shubs.io/how-i-bypassed-2-factor-authentication-on-google-yahoo-linkedin-and-many-others/

How to Bypass Two-Factor Authentication (2FA) and What the Future Holds (paper)

- September 30, 2014  |  By Etay Maor
• Security Intelligence

Chapter 7: Fundamentals -- Engineering a safer world: systems thinking applied to safety (book)

• Engineering a safer world: systems thinking applied to safety (book)
  pages 155 – 178

Chapter 19: Supporting Physicians by RE4S — Evaluating Requirements Engineering for Sustainability in the Medical Domain: Software Engineering for Sustainability (thesis)

• Software Engineering for Sustainability (thesis)
  pages 224 – 234
  published as:
  • Supporting physicians by RE4S: evaluating requirements engineering for sustainability in the medical domain
  • Birgit Penzenstadler, Joseph Mehrabi, and Debra J. Richardson
  • GREENS ’15 Proceedings of the Fourth International Workshop on Green and Sustainable Software, Pages 36-42
  • https://dl.acm.org/citation.cfm?id=2820166

Guidelines for conducting and reporting case study research in software engineering (paper)

• Per Runeson and Martin Höst
• Empirical Software Engineering
• April 2009, 14:131
• https://link.springer.com/article/10.1007%2Fs10664-008-9102-8

How The Economic Machine Works by Ray Dalio (talk)

https://www.youtube.com/watch?v=PHe0bXAluk0 30:59

On the (In)Security of Mobile Two-Factor Authentication (paper)

• Alexandra Dmitrienko, Christopher Liebchen, Christian Rossow, and Ahmad-Reza Sadeghi
• International Conference on Financial Cryptography and Data Security
• FC 2014: Financial Cryptography and Data Security pp 365-383
• https://link.springer.com/chapter/10.1007/978-3-662-45472-5_24

Overcoming Barriers to Medication Adherence for Chronic Diseases (talk)

https://www.youtube.com/watch?v=sEMCR7LchcA 1:00:52

ProtectID collecting a one-time-password via phone for Out Of Band Two-Factor Authentication (talk)

https://www.youtube.com/watch?v=2uADKDQE2ps 1:01

Requirements Engineering Usage Model - learning glass lecture 10 (talk)

https://www.youtube.com/watch?v=XuNxxolZUWo&list=PLUgFMzuE8lQDeixpbP3s6EyQx8P1NdeQL&index=15 7:46
for more details, see slides http://birgit.penzenstadler.de/teach/590/2015spring_CECS-590_10-UsageModel.pptx and https://www.slideshare.net/kamikitty/requirements-engineering-usage-models

Secure and Inclusive Authentication with a Talking Mobile One-Time-Password Client (paper)  

- Kristin Fuglerud; Oystein Dale  
- IEEE Security & Privacy (Volume: 9, Issue: 2, March-April 2011 ), Page(s): 27 - 34  

This is Paper C referred to in Inclusive design of ICT: The challenge of diversity (U Oslo thesis). There is a summary of it in 5.1.3 (pages 101 and 102).

Uses think-aloud protocol while users try system with followup interviews.

See comment on page 89 of thesis re making sure reviewers were clear about intent of paper (i.e., this is not a repeatable scientific experiment, but rather a qualitative exploration of an area).

SREcon18 Americas - Approaching the Unacceptable Workload Boundary (talk)

https://www.youtube.com/watch?v=8omFYif3J8 37:41  
slides: https://www.xaprb.com/slides/srecon-approaching-unacceptable-workload-boundary/#1

SREcon18 Americas - Whispers in Chaos: Searching for Weak Signals in Incidents (talk)

https://www.youtube.com/watch?v=UMyQ4EoPvC 39:30  
slides: https://www.slideshare.net/jpreed/whispers-in-chaos

discusses Trade-Offs Under Pressure: Heuristics and Observations Of Teams Resolving Internet Service Outages(thesis)

references SREcon18 Americas - Approaching the Unacceptable Workload Boundary (talk)  
Trade-Offs Under Pressure: Heuristics and Observations Of Teams Resolving Internet Service Outages(thesis)

- Allspaw, John  
- Division of Risk Management and Societal Safety  
- Lund University (2015), 87 pages  
- https://lup.lub.lu.se/student-papers/search/publication/8084520

The Use Case Technique - An Overview w/ Karl Wiegers (talk)

https://www.youtube.com/watch?v=MwimxkX5G5o 1:01:23  

Music for a Quiet Day

1. Andrew Staniland's Flute vs Cello (live) 7:07 https://www.youtube.com/watch?v=M1wL7_m3f-A  
2. 2CELLOS - Bach Double Violin Concerto in D minor (2nd movement) 6:49 https://www.youtube.com/watch?v=9s2fBvlGbT0  
3. Chris Cheek Berklee Quintet - Pelican Blues (Live at Berklee) 6:37 https://www.youtube.com/watch?v=iQ15cc7F4  
4. Haydn - Trio for flute, cello, and piano in G major, Hob. XV:15 20:06 https://www.youtube.com/watch?v=dCVaTYAnlOA  
5. Anoushka Shankar - Traces of You (Live) 11:02 https://www.youtube.com/watch?v=3E96xfgrU4  
6. Mozart: Clarinet Quintet in A Major, K.581: II. Larghetto · The TSO Chamber Soloists 7:01 https://www.youtube.com/watch?v=kcJI30mGmVU  
7. Hauschka - Familiar Things Disappear (Official Video) 5:34 https://www.youtube.com/watch?v=AXGP-8ck2ws  
8. Carola Bauchkolt, Laiufwerk 12:18 https://www.youtube.com/watch?v=5Pc92Zh_0W  
9. Mark Guiliana Jazz Quartet 'ABED', live at Band on the Wall 12:07 https://www.youtube.com/watch?v=zsyAgQ6P6ww  
10. Vivaldi, Michala Petri, Concerto Copenhagen and Lars Ulrik Mortensen 11:20 https://www.youtube.com/watch?v=_X6GJWcZD70
12. Javanese Gamelan Ensemble - Pelog Barang - Singa Nebah (The Pouncing Lion) 3:01 https://www.youtube.com/watch?v=p3HwqqiVxBc
13. Canon in D - Pachelbel 4:25 https://www.youtube.com/watch?v=mb9jibVF4
15. Jean-Baptiste Poyard - Telemann, violin fantasia n°9 5:04 https://www.youtube.com/watch?v=23JAcU331mg
20. Doctor of Music Arts Candidate Recital - Chai Chang Ning 9:13 https://www.youtube.com/watch?v=IA3zy20WxLY
22. Chris Thile - Bach: Sonata No. 1 in G Minor, BWV 1001 (Complete) 15:08 https://www.youtube.com/watch?v=j3H_TevSw5o
25. The Surroundings - Linda Catlin-Smith 8:15 https://www.youtube.com/watch?v=s5rhxZmTrY
26. Yo-Yo Ma never played it better than this. “Poem for Carlita” by Mark O’Connor 8:26 https://www.youtube.com/watch?v=Fhfb5jdz0uTQ
27. Performance by Ryogen | SingularityU Japan Summit 5:57 https://www.youtube.com/watch?v=EdW1wrdU6mR
28. Transported 16:04 https://www.youtube.com/watch?v=ud2eHawXM8B
29. The Knights - “Ascending Bird” 4:12 https://www.youtube.com/watch?v=1_iRpdL6LrE
31. “Ask” by Elliott Carter - Tanglewood Festival of Contemporary Music (copyright 2008) 12:08 https://www.youtube.com/watch?v=x7OGRXq2ZgC
32. Fables of Faubus by Charles Mingus performed by PROJECT Trio 3:59 https://www.youtube.com/watch?v=60eJV9y8bnQ
33. Liszt: Grandes Etudes De Paganini, S.141, No.3 In G-Sharp Minor “La Campanella” (Live F... 4:55 https://www.youtube.com/watch?v=-hNkp97130
35. Jean Rondeau records JC Bach Harpsichord Concerto In F Minor W 273 (Dynasty) 8:32 https://www.youtube.com/watch?v=SuNDBp4dnos
36. AMAZING: Mai K Namekawa plays Philip Glass Piano Etude No 9 & No 20 HD (dir. Andreas H. Bitesnich) 8:26 https://www.youtube.com/watch?v=H6UL4ZJ1t10
• Bach Brandenburg Concerto No.1 in F Major, BWV1046. Allegro / Adagio 7:58 https://www.youtube.com/watch?v=n24XxqOzsBY
• Bach Brandenburg Concerto No.1 in F Major, BWV1046. Menuet 12:08 https://www.youtube.com/watch?v=qV3Vl0y1EyE
• Bach - Brandenburg Concerto No 2 - Compass Rose Brass 10:03 https://www.youtube.com/watch?v=IRtxcLiLyY
• Johann Sebastian Bach - Brandenburg Concerto No.3 (BWV 1048) - Holland Baroque 10:37 https://www.youtube.com/watch?v=B1I_eboqB4
• J.S. Bach: Brandenburg Concerto No. 4 / Cappella Gabetta, Maurice Steger, Andrs Gabetta 14:40 https://www.youtube.com/watch?v=MafJXPWjkJ0
• J.S.Bach - Brandenburg Concerto No.5 in D BWV1050 - Croatian Baroque Ensemble 21:49 https://www.youtube.com/watch?v=BnjgAhFzs
• Bach's Brandenbourg Concerto No 6 performed by New York Baroque Incorporated and Monica Huggett 16:28 https://www.youtube.com/watch?v=ScwKWX30eoe
1. Welcome Performance: Jazzahead Group at TEDxUNC 11:04 https://www.youtube.com/watch?v=HjXOKSsZaFQ
2. Music from Bamboo, Music of Vietnam | The Bamboo Ensemble Suc Song Moi | TEDxBaDinh 11:23 https://www.youtube.com/watch?v=sF7KeOFgeC
4. Snarky Puppy, Metropole Orkest - The Curtain 14:45 https://www.youtube.com/watch?v=lJhBv-Aq2k2
5. Tigran Hamasyan - Fides Tua (Live) 5:22 https://www.youtube.com/watch?v=MAFJXPWjkJ0
8. Jean-Baptiste Poyard - Telemann, violin fantasia n°9 5:04 https://www.youtube.com/watch?v=23JAcU331mg
9. Mickey Freeman Music of Vietnam | The Bamboo Ensemble Suc Song Moi | TEDxBaDinh 11:23 https://www.youtube.com/watch?v=sF7KeOFgeC
10. Oscar Peterson's 'Easy Does It' - Robi Botos, piano, Dave Young, Bass, and Alvin Queen, Drums. 6:09 https://www.youtube.com/watch?v=J72xv6 computational mode
11. "In a sentimental mood" / Robi Botos Trio 11:35 https://www.youtube.com/watch?v=IRkg5bK4J6A
12. Tanj Modern Group Marga Luya - "Using Belek" 6:07 https://www.youtube.com/watch?v=il-hrbvz7M
13. Mahler Quartet In A minor for Piano, Violin, Viola, and Cello, Nicht zu schnell 9:06 https://www.youtube.com/watch?v=1QxTgQpDxME
15. The Venezuelan Project - Roots Medley (Live at Berklee) 7:53 https://www.youtube.com/watch?v=ZdBrGiYa6lc
16. Heifetz 2016: Stockhausen | Set Sail For The Sun 12:03 https://www.youtube.com/watch?v=J3H_TevSw5o
17. Adam Cockerham Plays Daniel Bacheler's "Paviv" on the Lute 5:04 https://www.youtube.com/watch?v=pelJBvBLZo
18. Avi Avital plays "Bucimis" (Traditional Bulgarian) 5:32 https://www.youtube.com/watch?v=xE0Ym3BFHfbw
20. Yudo - Pas De Trois 11:13 https://www.youtube.com/watch?v=m-OQyUv8Q
21. Lizotte - Raga 10:31 https://www.youtube.com/watch?v=0D2V-zWExs8
22. Gunther Schuller: Quartet for 4 Double Basses 16:42 https://www.youtube.com/watch?v=zdQsPSHS.uk
23. Appalachian Spring by Aaron Copland performed by Perspectives Ensemble 24:55 https://www.youtube.com/watch?v=DrYmM-LE2-OA
24. Theo Croker "This Could Be (For The Traveling Soul)" Live on Bside (BRIC TV) NYC 8:38 https://www.youtube.com/watch?v=kE0Yg3BHFbw
26. Occlus pro oculo totum orbem terrae caecat ("An eye for an eye makes the whole world blind") 17:58 https://www.youtube.com/watch?v=dxYpYoPLPnk
27. [NYCP] Lan-Ín Winnie Yang - Fantasy for Erhu and Strings (Andy Lin, erhu) 14:09 https://www.youtube.com/watch?v=V6SHQWClWoU
28. Thomas Adès: Arcadnia (Danish String Quartet) 20:12 https://www.youtube.com/watch?v=LeKa8Wk37RY
31. STREET MUSIQUE 8:46 https://www.youtube.com/watch?v=F-17Kd4Lyql&list=PLD0DF2A35516A8809&index=63
32. LORD OF THE RINGS - Sir James Galway - Philipp Jundt - KCO - Seoul 6:40 https://www.youtube.com/watch?v=oiayCQYOIO8
34. G.F. Handel: Concerto in B Flat Major HWV 325, Largo & Allegro; Voices of Music 4:41 https://www.youtube.com/watch?v=k5jiflRDIk
35. Schickele: "Ole Concerto" for Piano & Brass - Original Canadian Brass - Part 6 of 7 10:00 https://www.youtube.com/watch?v=mgITIHuyd8g
36. Beethoven: Violin Sonata No. 2 in A major (Ania Filochowska and Linzi Pan) 17:18 https://www.youtube.com/watch?v=7qRDC77Kxws
   - A Fifth Of Beethoven: 3:04 https://www.youtube.com/watch?v=W0JuLcSRqgH
d - PDQ Bach: "Erotica" Variations 7:36 https://www.youtube.com/watch?v=elSlpzEVjtg
   - 2017 Rock & Roll Hall of Fame Inductees ELO Perform "Roll Over Beethoven" 5:37 https://www.youtube.com/watch?v=S5H5ZSaLD7s
   - Eurythmics - Beethoven (I Love to Listen To) (Official Video) 5:26 https://www.youtube.com/watch?v=rBuMxyzouJQ
   - Orquestra Mineira de Rock - Because (The Beatles) 3:11 https://www.youtube.com/watch?v=3qjUW_3xDIQ
   - Beethoven's 5 Secrets - OneRepublic - The Piano Guys 5:31 https://www.youtube.com/watch?v=mJ_kw5j-t0

Notes past the eleventh class

- **Toward the final exam** 27 Nov 2017 7:42 pm now has json file for exam database as well as various pdf presentations of it. I don't expect to add any new questions to it. Later this week I may get a chance to proofread it and correct spelling/grammar issues, but primarily I plan only on making significant changes in response to difficulties with reading and/or understanding the questions and answers that get raised in the following week (before it goes to printing). If you have questions that require clarification, please contact me soonest to arrange makeup exam (same database, but different random key for question selection). Marks have to be handed in for people graduating this semester by the end of this semester's undergrad final period.

- **Allen Newell : Desires and Diversions : 1991 : Carnegie Mellon University** https://www.youtube.com/watch?v=sD4h29d1p1k
- Following Evaluating oral presentations (paper) as mentioned in Eighth class (Monday) 6 Nov 2017, marking scheme for presentations: Marking rubric for presentations.
- Regarding page count for final handin paper. The only discussion of this we have is in the proposal handin notes on Third Class (Monday) 25 Sept 2017 – so it hasn't changed since then.

Phoenix Pay example (new Canadian Federal payroll system circa 2016/2018)

While this kind of computer system stuff happens a lot, usually you just hear the end results like `Wells Fargo Accidentally Releases Trove of Data on Wealthy Clients` (https://www.nytimes.com/2017/07/21/business/dealbook/wells-fargo-confidential-data-release.html ). Unlike large corporations, when the government misses up, much more of what happened is visible and reported on. Since large government systems now incorporate significant computer components, there are many examples probably more significant than Phoenix pay, such as the online registration system rollout for Obamacare in the United States (cf. https://www.brookings.edu/blog/techtank/2015/04/09/a-look-back-at-technical-issues-with-healthcare.gov/ ). Beethoven's other classic: Violin Sonata (https://www.nytimes.com/2017/07/21/business/dealbook/wells-fargo-confidential-data-release.html ).

- Of interest on the topic is http://psacunion.ca/fix-phoenix the Fix Phoenix Paysystem! website of the PSAC (Public Service Alliance of Canada) which includes an archive of past announcements to members. Another useful source is The Professional Institute of the Public Service of Canada website on Phoenix Pay System (http://www.pipsc.ca/news-issues/phoenix-pay-system which also contains an archive of past announcements on problems with the system. On CBC, stories related to this are tagged "Phoenix Failing".}

- In August 2010, the Prime Minister announced that the Government of Canada will transform its pay administration.
• expected savings: 78.1 M per year
• Modern, commercial off-the-shelf pay system (PeopleSoft)
• System Integrator, IBM Canada, responsible for the design and implementation of the new pay system and processes.
• Phoenix will be deployed in three separate roll-outs in July, October, and December 2015
• Public Service Pay Centre opened in Miramichi, N. B. in March 2012.


• On budget on scope: schedule adjusted by 2 months
• Project completion date of April 2016
• Large quote (how things looked Nov 2015 before it would all go sideways – see item 10):
• Lessons learned to date:
  1) Strong governance and oversight: Hands-on oversight at the senior management levels facilitates responsive, decisive decision making to deliver project’s scope on time and within budget
  2) Strong negotiation skills: leveraged knowledge and expertise in leading negotiations with the private sector vendor to ensure outcomes are to the advantage of the Crown
  3) Strong financial management discipline: plan for the unexpected: Prudent management of funds by increasing the contingency envelope. As a result, the project is well prepared to mitigate risks, unforeseen issues & challenges.
  4) Clear accountabilities/responsibilities: clear accountabilities between Crown, vendor and key partners (i.e., Shared Service Canada for infrastructure; Office of the Chief Human Resources Officer for business processes, Canada School of Public Service for training) is essential.
  5) Strategic risk taking: Risk-focused governance approach results in the successful management and mitigation of the risks.
  6) Communication with stakeholders: a substantial level of communication is required to engage stakeholders, both internally and externally. The project initially underestimated the level and extent of required communication and subsequently adopted a more comprehensive engagement plan.
  7) Phased implementation: Phased implementation approach will help the project team learn from each successive phase to implement the next phase more effectively.
  8) Co-locating Crown and Vendor: Co-locating Crown and vendor teams enables greater information exchange between technical and business experts and allows more expedient issue resolution.
  9) Preparing departments for the transition: Departments require ongoing support to perform all preparations and transition activities. Departmental readiness is one of the keys to the success of the project.
• 10) Believe!


MP Weir on Phoenix Pay Problems (May 17, 2016) https://www.youtube.com/watch?v=DzbilkTy07k

MPs Weir & Mendicino Debate Phoenix Pay on CBC (July 18, 2016) https://www.youtube.com/watch?v=iv0_XyfrXdU

Phoenix pay system mess affects 80,000, government officials say (19 July 2016) http://www.cbc.ca/news/politics/phoenix-payroll-problems-fix-1.3683735


Phoenix pay system also breached federal workers’ privacy (The National, 19 July 2016) https://www.youtube.com/watch?v=iuuT_tADIR4

Phoenix pay system: Officials say all files will be processed by Oct. 31 (28 July 2016) http://globalnews.ca/news/2853127/phoenix-pay-system-officials-set-to-provide-update/


• quote: The contract to develop and operate the Phoenix payment system was awarded to IBM in 2011 after a lengthy RFP process that was only slightly shorter than the first F-35 jet fighter procurement process. At that time, Public Services and Procurement was impressed by IBM’s repeated assertion that its payroll software package could handle the complexities of paying federal government employees. As you undoubtedly know, their pay, benefit plans and pension contributions are governed by a myriad of union agreements made by the federal government with PSAC.
• quote: Beginning in early 2012 and continuing to today, IBM has not been shy about initiating many change orders to handle complexities that were not referenced in the original requirements documents issued by the federal government as part of the RFP process. The approved change orders have added millions in cost to the project and created delays as the project team worked hard to test all the added functionality.
• quote: Unbeknownst to us, Miramichi, NB is located beyond the edge of the known world as defined by the Internet backbone. As a result we have discovered that remotely located payroll advisers, which are rapidly increasing in number as described above, cannot effectively access the Phoenix system.


- "The story may ring eerily familiar to Australians after the recent census debacle: an IBM program, brought in to do a government job, collapsing in the face of its monumental new task (http://www.abc.net.au/news/2016-08-09/abs-website-inaccessible-on-census-night/7711652). It mirrors an earlier spectacular failure in Queensland, when IBM was contracted in 2007 to implement the government pay-roll system. Thousands of health workers suffered underpayment or no payment, and the cost to taxpayers is in the realm of $AU1.2 billion (http://www.abc.net.au/news/2016-07-27/qld-health-is-still-seeking-overpayments-six-years-on/7663230)."


Who should shoulder the blame for Phoenix fiasco? (21 Sept 2016) http://www.cbc.ca/news/canada/ottawa/phoenix-payroll-problems-ibm-1.3770947 -- Government has avoided laying blame on IBM for IT disaster, saying training the issue


** Phoenix Pay isn’t ‘Taking Care of Paychecks’ (‘Taking Care of Business’ parody) | 22 Minutes (CBC Comedy) https://www.youtube.com/watch?v=MJ4i-Ol5wOY (13 Nov 2016)

MP Weir on the Phoenix Pay System Boondoggle (Nov. 24, 2016) https://www.youtube.com/watch?v=US2WYLRdr7g

Phoenix pay system leaves Ottawa man broke while recovering from nearly fatal heart issue (7 Dec 2016) https://www.youtube.com/watch?v=kkMlntvBm1M

Phoenix Pay System (CTV news 14 Dec 2016, 34 minutes special) https://www.youtube.com/watch?v=RQvo1tz0VcY

Court order says government must work with unions to fix Phoenix (20 Jan 2017 ) http://psacunion.ca/court-order-says-government-must-work-unions-fix


Underpaid civil servants priority as payroll woes persist: Foote (7 Feb 2017) https://www.youtube.com/watch?v=j4cn6bh7mSA


MP Weir on the Phoenix Pay System’s 50,000 Incorrect T4s (Feb. 23, 2017) https://www.youtube.com/watch?v=QmPTy70W-Oc


Emergency pay available to employees on maternity, parental or disability leave and experiencing Phoenix pay problems (1 May 2017) http://psacunion.ca/emergency-pay-available-employees-maternity


Price tag for fixing Phoenix pay system now tops original cost -- Repair bill has risen to about $402M — more than $309.5M payroll implementation cost in the 1st place (24 May 2017) http://www.cbc.ca/news/canada/ottawa/phoenix-pay-update-may-24-1.4129049

Phoenix pay problems rise from the ashes amid summer hiring, new labour contracts -- The backlog affecting the public service pay system surged by about 10 per cent last month with new pay requests, officials say. (2 June 2017) https://www.thestar.com/news/canada/2017/06/02/phoenix-pay-problems-rise-from-the-ashes-amid-summer-hiring-new-labour-contracts.html


Federal scientists latest group burned by Phoenix pay problems -- 1,700 chemists and biologists underpaid or overpaid after new collective agreement (21 June 2017) http://www.cbc.ca/news/canada/ottawa/phoenix-pay-system-scientists-collective-agreement-1.4169959


No end in sight for Phoenix pay fiasco (July 24, 2017) http://thechronicleherald.ca/novascotia/1488695-no-end-in-sight-for-Phoenix-pay-fiasco


Nearly half of public servants paid by Phoenix have reported problems (8 Aug 2017) http://www.cbc.ca/news/canada/ottawa/phoenix-data-released-radio-canada-1.4259636 -- 156,035 employees up from 80,000 reported 19 July 2016 above -- also as of last month, 228,000 outstanding issues requiring resolution.


Western student forced to drop courses over Phoenix pay fiasco (22 Sept 2017) http://www.cbc.ca/news/canada/london/london-ontario-Phoenix-pay-western-student-1.4297247


Phoenix pay problems rise from the ashes amid summer hiring, new labour contracts -- The backlog affecting the public service pay system surged by about 10 per cent last month with new pay requests, officials say. (2 June 2017) https://www.thestar.com/news/canada/2017/06/02/phoenix-pay-problems-rise-from-the-ashes-amid-summer-hiring-new-labour-contracts.html


Federal scientists latest group burned by Phoenix pay problems -- 1,700 chemists and biologists underpaid or overpaid after new collective agreement (21 June 2017) http://www.cbc.ca/news/canada/ottawa/phoenix-pay-system-scientists-collective-agreement-1.4169959


No end in sight for Phoenix pay fiasco (July 24, 2017) http://thechronicleherald.ca/novascotia/1488695-no-end-in-sight-for-Phoenix-pay-fiasco


Nearly half of public servants paid by Phoenix have reported problems (8 Aug 2017) http://www.cbc.ca/news/canada/ottawa/phoenix-data-released-radio-canada-1.4259636 -- 156,035 employees up from 80,000 reported 19 July 2016 above -- also as of last month, 228,000 outstanding issues requiring resolution.


Western student forced to drop courses over Phoenix pay fiasco (22 Sept 2017) http://www.cbc.ca/news/canada/london/london-ontario-Phoenix-pay-western-student-1.4297247


Western student forced to drop courses over Phoenix pay fiasco (22 Sept 2017) http://www.cbc.ca/news/canada/london/london-ontario-Phoenix-pay-western-student-1.4297247


Summary quote: "Overall, we found that there was no oversight of the Phoenix project, which allowed Phoenix executives to implement the system even though they knew it had significant problems. There were no oversight bodies independent of the project management structure to provide independent advice to the Deputy Minister of Public Services and Procurement on the project status. This meant that the Deputy Minister did not receive independent information showing that Phoenix was not ready to be implemented or that the Miramichi Pay Centre and departments and agencies were not ready for Phoenix. Phoenix executives were more focused on meeting the project budget and timeline than on what the system needed to do."

Phoenix problems evident, but not brought forward due to work culture, auditor general says (CBC 28 May 2018) [http://www.cbc.ca/news/politics/auditor-general-first-nations-phoenix-1.4681172](http://www.cbc.ca/news/politics/auditor-general-first-nations-phoenix-1.4681172)


**Searching the Computer Science Literature**

My favourite way to search is using [https://scholar.google.ca/](https://scholar.google.ca/)

For example I can search for `go playing programs` and one of the hits I get is:

- **Mastering the game of Go with deep neural networks and tree search**

  - D Silver, A Huang, CJ Maddison, A Guez, L Sifre... - Nature, 2016 - [nature.com](http://nature.com)

  ... a, Plot showing the playing strategy of policy networks as a function of their training ... a, Results of a tournament between different Goprocesses (see Extended Data Tables 6,7,8,9,10,11 ... To provide a greater challenge to AlphaGo, some programs (pale upper bars) were given four ... Cited by 1443 Related articles All 45 versions Web of Science: 275 CiteSave

The top link gets me to the copy at Nature. The link on D Silver gets me to a page on the publications of David Silver (who works at DeepMind, Google). It is impressive that this 2016 paper has already been cited 1443 times, and the Cited by link gets me there. The 45 versions list gives, among other things, a link at Research Google [https://research.google.com/pubs/pub44806.html](https://research.google.com/pubs/pub44806.html) on the paper that provides an abstract and then sends you to Nature for the actual paper.

The other thing worth noting in the original google scholar search result was in the left margin was a mechanism to set the time range for publication dates in the search. This capability is also in the citation list that the Cited by link produces.

One can use google scholar to search for theses with queries like `go playing programs "In this thesis"`. This works because in writing a couple hundred pages of thesis, most people at some point use the phrase `In this thesis` to start a sentence. Some times you will find it in a paper where material has been cut and pasted from a thesis (or where the author is referring to a referenced thesis rather than the paper itself). Actually the above query didn't match well, so I tried the more restrictive `"In this thesis" "game of go"` and got some relevant theses. One was about modelling spiking neurons at: [https://search.proquest.com/docview/1903600006?pq-origsite=gscholar](https://search.proquest.com/docview/1903600006?pq-origsite=gscholar). From the URL, you can see it is at ProQuest, which is something we have access to via UWO digital library subscriptions. Other theses we see being put up by their home universities, like UWO does, giving us a list of specific repositories that can be searched for free:

- [https://uwspace.uwaterloo.ca/handle/10012/12173](https://uwspace.uwaterloo.ca/handle/10012/12173)
- [https://depositonce.tu-berlin.de/handle/11303/6507](https://depositonce.tu-berlin.de/handle/11303/6507)
- [http://eprints.nottingham.ac.uk/43398/](http://eprints.nottingham.ac.uk/43398/)
- [http://archiv.ub.uni-heidelberg.de/volltextserver/22742/](http://archiv.ub.uni-heidelberg.de/volltextserver/22742/)
- [https://dspace.mit.edu/handle/1721.1/108982](https://dspace.mit.edu/handle/1721.1/108982)
- [http://ir.library.oregonstate.edu/xmlui/handle/1957/60635](http://ir.library.oregonstate.edu/xmlui/handle/1957/60635)

Some other repositories are:

- [https://dspace.library.uu.nl/](https://dspace.library.uu.nl/)
- [http://ir.lib.uwo.ca/etd/](http://ir.lib.uwo.ca/etd/)
- [https://tspace.library.utoronto.ca/handle/1807/9944](https://tspace.library.utoronto.ca/handle/1807/9944)
- [https://qspace.library.queensu.ca/handle/1974/290](https://qspace.library.queensu.ca/handle/1974/290)
- [https://www.repository.cam.ac.uk/](https://www.repository.cam.ac.uk/)
Similarly, rather than searching all of google scholar, you can target searches in specific publishers that cover computer science, the largest being:

- [http://dl.acm.org/](http://dl.acm.org/)
- [http://ieeexplore.ieee.org/Xplore/home.jsp](http://ieeexplore.ieee.org/Xplore/home.jsp)
- [https://link.springer.com/](https://link.springer.com/)

Note that each search area has its own undocumented interface that needs to be poked and prodded to yield its secrets.

It is also worth noting that sometimes a simple google search will find papers that aren't in google scholar. Among other reasons, authors tend to put up copies of their work and their home pages don't seem to get indexed by google scholar. Also, it can sometimes be interesting to search for a topic in google by searching for videos or blogs such as

- [https://www.google.ca/search?q=go+playing+program+youtube&oq=go+playing+program+youtube](https://www.google.ca/search?q=go+playing+program+youtube&oq=go+playing+program+youtube)
- [https://www.google.ca/search?q=go+playing+program+vimeo&oq=go+playing+program+vimeo](https://www.google.ca/search?q=go+playing+program+vimeo&oq=go+playing+program+vimeo)
- [https://www.google.ca/search?q=go+playing+program+blog&oq=go+playing+program+blog](https://www.google.ca/search?q=go+playing+program+blog&oq=go+playing+program+blog)


While the page itself has some good information, the real material is toward the end where you see sections entitled See Also, Notes, References (155 sources), Sources, Further Reading, and External Links. And, throughout the article are links to more computer related Go material: [https://en.wikipedia.org/wiki/Computer_Go](https://en.wikipedia.org/wiki/Computer_Go), [https://en.wikipedia.org/wiki/Go_and_mathematics](https://en.wikipedia.org/wiki/Go_and_mathematics), [https://en.wikipedia.org/wiki/AlphaGo](https://en.wikipedia.org/wiki/AlphaGo), [https://en.wikipedia.org/wiki/Internet_Go_Server](https://en.wikipedia.org/wiki/Internet_Go_Server), and [https://en.wikipedia.org/wiki/Smart_Game_Format](https://en.wikipedia.org/wiki/Smart_Game_Format) etc.

**Thinking about systems**

- Intros to the systems approach
  - The Contribution of the Systems Sciences to the Humanities (paper) – Ervin Lazlo
  - A Definition of Systems Thinking: A Systems Approach (paper)
  - In A World of Systems (video) a short overview video on viewing the world as systems
  - Introduction to System Dynamics: Overview (talk) (John Sturma's introduction to the MIT course on System Dynamics)
    - Introduction to System Dynamics (MIT OpenCourseWare page)
  - The Common Foundation Underlying Physical and Social Systems - Jay W. Forrester (talk) (Jay Forrester was one of the founders of systems dynamics study)
    - Applications of System Dynamics - Jay W. Forrester (talk)
    - Jay Forrester: The Growth and Collapse of Markets (talk)
  - A Philosophical Look at System Dynamics (talk) an overview talk from a course on Systems Dynamics by Donella Meadows (who was also involved in the work on general sustainability by the Club of Rome (cf The Club of Rome and Limits To Growth: Achieving the Best Possible Future (talk)) )
    - Other talks from the same course:
      - Creating Models from Scratch (talk)
      - Systems: Overshoot and Collapse (talk)
      - System Dynamics Model: Kaibab Deer Population (talk)
  - An introduction to critical systems (talk) (Ian Sommerville talk that brings in requirements to discussion of systems)
    - Critical National Infrastructure (talk)
    - Infrastructure dependability (talk)
    - Infrastructure resilience (talk)
    - Critical systems engineering (talk)
    - Introducing sociotechnical systems (talk)
    - System success and failure (talk)
    - Emergent properties of sociotechnical systems (talk)
    - Systems of systems (talk)
- More on Socio-Technical Systems
  - Towards Acceptance of Socio-technical Systems – An Emphasis on the Requirements Phase (paper)
  - The Santa Fe Institute has made a specialty of studying complex systems, e.g.,
    - Norms that Govern: the Case of Wikipedia (talk)
    - Ties that Bind: The Goodness of Social Networks (talk)
    - From Scaling to Practice: Real World Applications of Cities Research (talk)
    - The Complex Science of Cyberdefense: Computer Immunology (talk)
    - Software Engineering: Evolving Computer Programs (talk)
    - The Myth of Common Sense: Why Everything that Seems Obvious Isn't (talk)

**A Definition of Systems Thinking: A Systems Approach (paper)**
An introduction to critical systems (talk)

https://www.youtube.com/watch?v=msp05wJ7fD4  9:48

Critical National Infrastructure (talk)

https://www.youtube.com/watch?v=Z6xtLh5DkbE  8:22

Critical systems engineering (talk)

https://www.youtube.com/watch?v=gLNGWvu8U3k  11:28

Emergent properties of sociotechnical systems (talk)

https://www.youtube.com/watch?v=ZCBaQpEq1U8  9:06

slides: https://www.slideshare.net/sommerville-videos/emergent-properties

Infrastructure dependabilty (talk)

https://www.youtube.com/watch?v=w209SCYioUo  18:08

Infrastructure resilience (talk)

https://www.youtube.com/watch?v=z7Cwtpmj_VY  11:55

Introducing sociotechnical systems (talk)

https://www.youtube.com/watch?v=xdFftbIToV0  9:20

slides: https://www.slideshare.net/sommerville-videos/introducing-sociotechnical-systems

Systems of systems (talk)

https://www.youtube.com/watch?v=ryLeFaHarPQ  6:45
System success and failure (talk)

https://www.youtube.com/watch?v=VkW7Dr3SXkg  11:42

A Philosophical Look at System Dynamics (talk)

https://www.youtube.com/watch?v=XL_JOoomRTA  53:18

Creating Models from Scratch (talk)

https://www.youtube.com/watch?v=4Hjy_wFMy_g  42:46

System Dynamics Model: Kaibab Deer Population (talk)

https://www.youtube.com/watch?v=2rUXm5b-gZM  53:40

for a short talk with better visuals on the Kaibab Deer model, see Deer population model in VENSIM (talk) which explains how to generate visuals in Sustainability Modeling Spring 2013 Lecture 6: Modeling Population-Nature Interactions (talk)


see also discussion in https://en.wikipedia.org/wiki/Kaibab_Plateau

see also discussion in http://ccl.northwestern.edu/netlogo/models/WolfSheepPredation and http://ccl.northwestern.edu/papers/wolfsheep.pdf

Deer population model in VENSIM (talk)

https://www.youtube.com/watch?v=QVausXIJP-U  16:29

Sustainability Modeling Spring 2013 Lecture 6: Modeling Population-Nature Interactions (talk)

https://www.youtube.com/watch?v=OVpMBQONZM4  31:31

Systems: Overshoot and Collapse (talk)

https://www.youtube.com/watch?v=f9g4-5-GKBc  1:02:21

The Contribution of the Systems Sciences to the Humanities (paper)

Ervin Laszlo and Alexander Laszlo

In A World of Systems (video)
Introduction to System Dynamics: Overview (talk)

Introduction to System Dynamics (MIT OpenCourseWare page)

Norms that Govern: the Case of Wikipedia (talk)

From Scaling to Practice: Real World Applications of Cities Research (talk)

Software Engineering: Evolving Computer Programs (talk)

The Complex Science of Cyberdefense: Computer Immunology (talk)

The Myth of Common Sense: Why Everything that Seems Obvious Isn't (talk)

Ties that Bind: The Goodness of Social Networks (talk)

The Common Foundation Underlying Physical and Social Systems - Jay W. Forrester (talk)

Applications of System Dynamics - Jay W. Forrester (talk)
Understanding humans

Functional requirements generally work in terms of a self-contained world of one piece of code interacting with another. Non-functional requirements stem from the notion that the software is part of a system. Ultimately, such systems are parts of larger systems that include humans as components. Having the system work properly often requires a notion of how humans behave within the system. Safety requirements often want to make the system safe for humans. Security requirements often want to make the system secure for humans against other humans. Usability and accessibility are generally understood to be usable and accessible for humans. Sustainability often involves creating a system that will support the efforts of an organization or the human race in general to sustain. One way to test to see if these requirements are met is to perform experiments involving humans. The other approach is to have some fundamental understanding of the role humans play in systems and reason about what should be done. An intermediate approach is to use simulations of humans.

One model of human behavior that has been used in requirements analysis is ACT-R, which supports simulation of human components in a system from a cognitive level view.

- ACT: A simple theory of complex cognition (paper)
- ACT-R system home page (website)

Another model of human behavior that has also been used to investigate human cognitive behavior in systems is SOAR

- SOAR system home page (web site)

An interesting way to understand humans is by looking at the kinds of errors they make

- The role of error in organizing behaviour (paper)
- A Classification Model for Human Error in Collaborative Systems (thesis)

Instead of thinking about people as information processors, another approach is to understand people through what motivates them. One such theory is Self Determination Theory

- Self-Determination Theory (website)

In addition to how people think, another aspect of human components in systems that is important is why they make errors. One approach to improving error rates is checklists:

- Electronic Checklist Support for Disaster Response (paper)
- The Use and Design of Flightcrew Checklists and Manuals (report)
- Semantic services for information and management support in mass casualty incident scenarios (paper)

A Classification Model for Human Error in Collaborative Systems (thesis)

- David Trepess

- Alexander, Tiffaney,
  - http://stars.library.ucf.edu/etd/5622

The role of error in organizing behaviour (paper)

- J Rasmussen
  - This is a reprint of a paper that appeared in Ergonomics 1990, Volume , pages–99 Copyright © 1990 Taylor & Francis Ltd.
  - http://qualitysafety.bmj.com/content/12/5/377.short
  - https://pdfs.semanticscholar.org/88c1/e42bd5898022b0092acbf4cc6c92d8dec017.pdf

ACT: A simple theory of complex cognition (paper)

- Anderson, J. R.

ACT-R system home page (website)

http://act-r.psy.cmu.edu/


- Jill Fain Lehman, John Laird, and Paul Rosenbloom
  - 37 pages.

SOAR system home page (web site)

https://soar.eecs.umich.edu/

Electronic Checklist Support for Disaster Response (paper)

- U Krüger, F Wucholt, C Beckstein
  - Proceedings of the 9th International ISCRAM Conferences 2012, 5 pages

Semantic services for information and management support in mass casualty incident scenarios (paper)

- Uwe Krüger, Aygul Gabdulkhakova, Birgitta König-Ries, and Clemens Beckstein
  - ServiceWave’10 Proceedings of the 2010 international conference on Towards a service-based internet
The Use and Design of Flightcrew Checklists and Manuals (report)

- John W. Turner and M. Stepheb Huntley, Jr.
- 1991
- 71 pages.

Self-Determination Theory (website)

http://selfdeterminationtheory.org/

Overview of non-functional software requirements and requirements engineering

One can start with the general topic of requirements engineering

- An introduction to Requirements Engineering (talk) an Ian Sommerville talk introducing the topic of requirements engineering
- Requirements Engineering (book)
- Requirements engineering: a good practice guide (book)
- Four dark corners of requirements engineering (paper)
- Experimenting with Error Abstraction in Requirements Documents (paper)
- Requirements in the wild: How small companies do it (talk)

Of course all systems fail, so a part of system requirements is how failures should be resolved. For socio-technical systems, this is `emergency response'.

- Investigation of user requirements in the emergency response sector: the Dutch case (paper)
- Disaster Robotics (book)
- User requirements for a collective intelligence emergency response system (paper)
- Semantic Representation of Process and Service Compliance – A Case Study in Emergency Planning (paper)
- Design and evaluation of mass evacuation support systems using ontologies for improved situation awareness (Massey PhD Thesis)
- GDIA: Eliciting information requirements in emergency first response (paper)
- Emergency Scenario User Perspective in Public Safety Communications Systems (paper)
- Information systems architecture for fire emergency response (paper)
- Developing Effective Standard Operating Procedures - Guide To Developing Effective Standard Operating Procedures For Fire And EMS Departments (report)
- Evaluating the Reliability of Emergency Response Systems for Large-Scale Incident Operations (report)

Requirements very firmly sit at the boundary between formal computer stuff and fuzzy people stuff. This raises a number of issues, such as:

- Getting to the Shalls: Facilitating Sensemaking in Requirements Engineering (paper) more detail on requirements engineering
- Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study (paper) a look at issues in getting system component requirements right
- Usability of requirements techniques: a systematic literature review (paper)

Nonfunctional requirements tend to lean even further in the fuzzy people side although there are still ways to organize what is going on

- On Non-Functional Requirements (paper)
- Automated classification of non-functional requirements (paper)
- Quality Requirements in Agile as a Knowledge Management Problem: More than Just-in-Time (paper)

One frequently used approach is `goal-based'

- Goal-Based Requirements Analysis (paper)
- Goal Identification and Refinement in the Specification of Information Systems (Georgia Tech thesis)
- Representing and Using Nonfunctional Requirements: A Process-Oriented Approach (paper)

Interestingly enough, there is actually some formal analysis of interest regarding how to deal with the preferences of multiple stakeholders

- Voting Systems and the Condorcet Paradox | Infinite Series (talk)
Some theses focused on requirements engineering:

- Missing Requirements Information and its Impact on Software Architectures: A Case Study (UWO Thesis)
- Maps of Lessons Learnt in Requirements Engineering (UWO Thesis)
- GalusT: supporting the extraction of rights and obligations for regulatory compliance (paper)
- Regulatory Compliance-oriented Impediments and Associated Effort Estimation Metrics in Requirements Engineering for Contractual Systems Engineering Projects (UWO Thesis)
- The Prevalence and Impact of Persistent Ambiguity in Software Requirements Specification Documents (U Waterloo thesis)
- Extracting Non-Functional Requirements from Unstructured Text (U Waterloo thesis)
- The Impact of Domain Knowledge on the Effectiveness of Requirements Engineering Activities (U Waterloo Thesis)

And relevant journals:

- Requirements Engineering (journal)

Are "Non-functional" Requirements really Non-functional? An Investigation of Non-functional Requirements in Practice (paper)

- Jonas Eckhardt; Andreas Vogelsang; Daniel Méndez Fernández
- Software Engineering (ICSE), 2016 IEEE/ACM 38th International Conference on
- 11 pages.

Automated classification of non-functional requirements (paper)

- Jane Cleland-Huang, Raffaela Settimi, Xuchang Zou and Peter Solc
- Requirements Engineering
- Volume 12 Issue 2, May 2007
- Pages 103 - 120
- https://link.springer.com/article/10.1007%2Fs00766-007-0045-1

Design and evaluation of mass evacuation support systems using ontologies for improved situation awareness (Massey PhD Thesis)

- Javed, Yasir
- Information Sciences at Massey University, Albany, New Zealand
- 2012, 298 pages.
- https://mro.massey.ac.nz/handle/10179/4045

Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study (paper)

- Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study
- Wenhua Hu, Jeffrey C. Carver, Vaibhav K. Anu, Gursimran S. Walia and Gary Bradshaw
- http://dl.acm.org/citation.cfm?doid=2961111.2962596

Experimenting with Error Abstraction in Requirements Documents (paper)

- Experimenting with Error Abstraction in Requirements Documents
- Filippo Lanubile, Forrest Shull, and Victor R. Basili
- METRICS '98 Proceedings of the 5th International Symposium on Software Metrics, 8 pages.

Extracting Non-Functional Requirements from Unstructured Text (U Waterloo thesis)
Four dark corners of requirements engineering (paper)

- Pamela Zave and Michael Jackson
- ACM Transactions on Software Engineering and Methodology (TOSEM) TOSEM Homepage archive
- Volume 6 Issue 1, Jan. 1997
- Pages 1-30
- https://dl.acm.org/citation.cfm?doid=237432.237434

GaiusT: supporting the extraction of rights and obligations for regulatory compliance (paper)

GaiusT: supporting the extraction of rights and obligations for regulatory compliance
Nicola Zeni, Nadzeya Kiyavitskaya, Luisa Mich, James R. Cordy, and John Mylopoulos

GDIA: Eliciting information requirements in emergency first response (paper)

- Lili Yang, Raj Prasanna, and Malcolm King
- Requirements Engineering
- November 2015, Volume 20, Issue 4, pp 345–362

Developing Effective Standard Operating Procedures - Guide To Developing Effective Standard Operating Procedures For Fire And EMS Departments (report)

- IOCAD Emergency Services Group Inc.
- 103 pages.

Emergency Scenario User Perspective in Public Safety Communications Systems (paper)

- Berrouard, D., Cziner, K. and Boukalov. A.
- http://www.iscram.org/legacy/ISCRAM2006/ISCRAM2006Proceedingszip/PapersTuesday/S3_T1_5_Berrouard_etal.pdf

Evaluating the Reliability of Emergency Response Systems for Large-Scale Incident Operations (report)

- by Brian A. Jackson, Kay Sullivan Faith, Henry H. Willis
- RAND Health Quarterly, Fall 2012
- https://www.rand.org/pubs/monographs/MG994.html
- 225 pages.

Information systems architecture for fire emergency response (paper)
Getting to the Shalls: Facilitating Sensemaking in Requirements Engineering (paper)

- Getting to the Shalls: Facilitating Sensemaking in Requirements Engineering
- Suranjan Chakraborty, Christoph Rosenkranz, and Josh Dehlinger
- ACM Transactions on Management Information Systems (TMIS) - Special Issue on Complexity of Systems Evolution: Requirements Engineering Perspective
- Volume 5 Issue 3, January 2015, Article No. 14, 30 pages.
- http://dl.acm.org/citation.cfm?doid=2666081.2629351

Goal-Based Requirements Analysis (paper)

- Annie I. Antón

Goal Identification and Refinement in the Specification of Information Systems (Georgia Tech thesis)

- Annie I. Antón
- Ph.D. Thesis, Georgia Institute of Technology, June 1997

Representing and Using Nonfunctional Requirements: A Process-Oriented Approach (paper)

- J. Mylopoulos, L. Chung and B. Nixon
- IEEE Transactions on Software Engineering - Special issue on knowledge representation and reasoning in software development
- Volume 18 Issue 6, June 1992
- Page 483-497

Investigation of user requirements in the emergency response sector: the Dutch case (paper)

- Second International Symposium on 'Geo-information for disaster management', 2006. 6 pages.
- http://library.wur.nl/WebQuery/wurpubs/351958

Disaster Robotics (book)

- By Robin R. Murphy
- Paperback, 240 pp.
- https://mitpress.mit.edu/books/disaster-robotic
Missing Requirements Information and its Impact on Software Architectures: A Case Study (UWO Thesis)

- Missing Requirements Information and its Impact on Software Architectures: A Case Study
- Md Rounok Salehin, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1811/

Maps of Lessons Learnt in Requirements Engineering (UWO Thesis)

- Maps of Lessons Learnt in Requirements Engineering
- Ibtehal Noorwali, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/3996/

Regulatory Compliance-oriented Impediments and Associated Effort Estimation Metrics in Requirements Engineering for Contractual Systems Engineering Projects (UWO Thesis)

- Regulatory Compliance-oriented Impediments and Associated Effort Estimation Metrics in Requirements Engineering for Contractual Systems Engineering Projects
- Md Rashed I. Nekvi, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/4319/

Quality Requirements in Agile as a Knowledge Management Problem: More than Just-in-Time (paper)

- Eric Knauss; Grischa Liebel; Kurt Schneider; Jennifer Horkoff; Rashidah Kasauli
- Requirements Engineering Conference Workshops (REW), 2017 IEEE 25th International, 4 pages.

Requirements and testing: seven missing-link myths (paper)

- D. Graham
- Page(s): 15 - 17

Requirements engineering: a good practice guide (book)

- Requirements engineering: a good practice guide
- by Sommerville, Ian; Sawyer, Pete
- 1997
- IN LIBRARY, QA76.76.D47S668 1997, Taylor Library Stack 6 (S6) - Regular Loan
- http://alpha.lib.uwo.ca/record=b2612622
- https://www.amazon.ca/Requirements-Engineering-Good-Practice-Guide/dp/0471974447

An introduction to Requirements Engineering (talk)
Requirements engineering: a roadmap (paper)

- Bashar Nuseibeh and Steve Easterbrook
- ICSE '00 Proceedings of the Conference on The Future of Software Engineering
- Pages 35-46
- https://dl.acm.org/citation.cfm?doid=336512.336523

Requirements Engineering (book)

- Jeremy Dick, Elizabeth Hull, Ken Jackson
- Springer, 2017

Requirements Engineering (journal)

Requirements in the wild: How small companies do it (talk)

https://www.youtube.com/watch?v=LO_J7mhFwnY  53:12

- quote: In the US in 2002, 95% of all software firms have < 50 employees, 21% of total income in field; 28% of employees.
- studied companies with offices in Toronto
- lesson 1: everyone does requirements differently
- hypothesis: companies are different due to natural selection of business competition in their niche
- lesson 2: strong cultural cohesion
- hypothesis: in small companies, efficient team dynamics is more important than good process
- end goal is ‘shared understanding’. a good team does this easily, a good process might help a weak team
- lesson 3: the CEO is the requirements engineer
- hypothesis: being a good entrepreneur may require one to be good at figuring out requirements
- hypothesis: in small companies, business strategy and requirements engineer are tightly interconnected
- lesson 4: requirement errors are not catastrophes
- hypothesis: perhaps these companies are doing minor variations on what they already know
- radical design versus normal design
- hypothesis: perhaps a small company can more easily resolve problems that arise because everyone knows each other and what is going on
- hypothesis: perhaps a catastrophe would kill a small company, so it wouldn't be around to be interviewed
- successful small companies are risk adverse and avoid significant change in process
- a next step would be to try and test the hypothesis to see if they are right
- software engineering should address what sort of business situation a particular approach is appropriate for
- processes seem to become more important as companies get larger

The Prevalence and Impact of Persistent Ambiguity in Software Requirements Specification Documents (U Waterloo thesis)

- Ribeiro, Cristina
- 2016-12-21
- https://uwspace.uwaterloo.ca/handle/10012/11123

The Impact of Domain Knowledge on the Effectiveness of Requirements Engineering Activities (U Waterloo Thesis)
Usability of requirements techniques: a systematic literature review (paper)

- Usability of requirements techniques: a systematic literature review
- Denise Bombonatti, Catarina Grahia, Ana Moreira, João Araújo and Miguel Goulão
- SAC ‘16 Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pages 1270-1275
- http://dl.acm.org/citation.cfm?doid=2851613.2851758

User requirements for a collective intelligence emergency response system (paper)

- Vita Lanfranchi and Neil Ireson
- BCS-HCI ’09 Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology
- Pages 198-203
- https://dl.acm.org/citation.cfm?id=1671035

Semantic Representation of Process and Service Compliance – A Case Study in Emergency Planning (paper)

- Aygul Gabdulkhakova, Birgitta König-Ries, and Norris Syed Abdullah
- CAiSE 2011: Advanced Information Systems Engineering Workshops pp 249-258
- https://link.springer.com/chapter/10.1007/978-3-642-22056-2_27

Voting Systems and the Condorcet Paradox | Infinite Series (talk)

https://www.youtube.com/watch?v=HoAnYQZrNrQ 11:35

From requirements to design

One view of the connection of requirements to design is:

- From Non-Functional Requirements to Design through Patterns (paper)

From Non-Functional Requirements to Design through Patterns (paper)

- Daniel Gross and Eric Yu
- Requirements Engineering
- February 2001, Volume 6, Issue 1, pp 18–36

How safety requirements figure into software systems
An intro:

- Safety-Critical Systems - Professor Martyn Thomas CBE (talk)
- Air Force System Safety Handbook (report)
- DevOpsDays Detroit 2017: Day 1 Keynote by Dr. Richard I. Cook (video)
- Safe operation as a social construct (paper)
- Is safety a subject for science? (paper)

It is worth considering whether or not it meets basic professional ethics standards to produce unsafe systems

- The ethics of safety-critical systems (paper)
- ACM Code of Ethics

The interrelation between safety and requirements engineering (Overview of non-functional software requirements and requirements engineering)

- Engineering safety-related requirements for software-intensive systems (paper) – A tutorial taking the point of view that requirements engineers need to know more about safety engineering and safety engineers need to know more about requirements engineering
- Integration between requirements engineering and safety analysis: A systematic literature review (paper)
- Requirements engineering for safety-critical systems: A systematic literature review (paper)

As always with requirements, there is the question of how do you know if it is satisfied or not

- Towards state-of-the-art and future trends in testing of active safety systems (paper)
- Verification of safety-critical software (paper)
- The Safety of Software – Constructing and Assuring Arguments (U of York thesis)

Peter G Neumann https://en.wikipedia.org/wiki/Peter_G._Neumann has been collecting news items of failures of systems incorporating computers for a long time with his Risks Digest dating back to 1985 http://catless.ncl.ac.uk/Risks/

- Inside risks: a risks-related bookshelf (paper)

Nancy Leveson https://en.wikipedia.org/wiki/Nancy_Leveson is a researcher in safety noted for advocating a systems (Thinking about systems) approach for both safety and security (Security as a non-functional software system requirement) concerns.

- An Approach to Designing Safe Embedded Software (paper)
- A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on STPA. (paper)
- Engineering financial safety: a system-theoretic case study from the financial crisis (MIT thesis)
- Software Challenges In Achieving Space Safety (paper)
- An integrated approach to safety and security based on systems theory (paper)
- Early Concept Development and Safety Analysis of Future Transportation Systems (paper)
- Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems (paper)
- The Need for a Paradigm Shift in Safety and Cyber Security (talk)
- Introduction to: Systems Theoretic Accident Model & Processes (STAMP) WEBINAR REPLAY (talk)
- Engineering a safer world: systems thinking applied to safety (book)
- Designing and Operating Safety Systems: The Missing Link - SDM alumnus John Helferich (talk)

Other references to the connection between safety and security

- The SEMA referential framework: Avoiding ambiguities in the terms “security” and “safety” (paper)

A different, but related, approach is to view the issue as a risk management problem

- Coping with Defective Software in Medical Devices (paper)

Fault-tree analysis is an older safety engineering approach

- Fail-safe testing of safety-critical systems: a case study and efficiency analysis (paper) – fail-safe is the notion that when failure happens, you don't want it to make matters worse – for example, by default elevator breaks are engaged and it takes power to have them retract so that the elevator can move https://en.wikipedia.org/wiki/Fail-safe

Different from fault-tree analysis is the technique for testing reliability of systems known as ‘fault injection’

- Assessing Dependability with Software Fault Injection: A Survey (paper)
- Why do some (weird) people inject faults? (paper)

One can also get a better view of the safety issues in an existing system through accident reports

- Drivers' needs and safety systems (paper)

More generally, one can ask what is the role of human components in the overall safety of a system

- Success and failure: human as hero – human as hazard (paper)
People interested in software quality (perhaps the ultimate non-functional requirement) improvement generally focus on the software development process rather than the software end product. Similarly, one can say that building software meeting a safety requirement requires a development software aimed at meeting such a requirement.

- **Analysing the Safety of a Software Development Process (paper)**
- **Opening the sources of accountability (paper)**
- **Building a safety architecture pattern system (paper)**

Some system examples

- **Design, Implementation, and Verification of a Reactor Protection System Using HFC6000 (UWO Thesis)**—probabilistic safety margin analysis
- **Predictive Shutdown Systems for Nuclear Power Plants (UWO Thesis)**—improve safety margin by monitoring system to predict probable encounter with a safety threshold

The notions of risk and safety are closely related. One might equally ask if a technology is safe or if it is risky. Current technologies that pose this concern are nanotechnology and artificial intelligence. Past technologies would include bio-chemical weapons and nuclear technology.

- **Safety Issues in Advanced AI (talk)**
- **Facing the unknown: the future of humanity (talk)**

Conferences:

- **International Conference on Computer Safety, Reliability, and Security (proceedings)**

**Air Force System Safety Handbook (report)**

- Air Force Safety Agency
- revised July 2000
- 161 pages.

**Analysing the Safety of a Software Development Process (paper)**

- Analysing the Safety of a Software Development Process
- Stephen Paynter and Bob W. Born
- SAFECOMP '02 Proceedings of the 21st International Conference on Computer Safety, Reliability and Security, Pages 186-197
- [https://link.springer.com/chapter/10.1007/3-540-45732-1_19](https://link.springer.com/chapter/10.1007/3-540-45732-1_19)

**An Approach to Designing Safe Embedded Software (paper)**

- An Approach to Designing Safe Embedded Software
- Nancy G. Leveson
- [https://link.springer.com/chapter/10.1007/3-540-45828-X_2](https://link.springer.com/chapter/10.1007/3-540-45828-X_2)

**Designing and Operating Safety Systems: The Missing Link - SDM alumnus John Helferich (talk)**

[https://www.youtube.com/watch?v=4hqHrqKvbXs](https://www.youtube.com/watch?v=4hqHrqKvbXs) 59:21


**Engineering a safer world: systems thinking applied to safety (book)**

Engineering a safer world: systems thinking applied to safety
by Leveson, Nancy
Engineering systems, 2011
Book:
IN LIBRARY, T55.L466 2011, Taylor Library Stack 5 (S5) - Regular Loan
A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on STPA. (paper)

- Abdulkhaleq, Asim, Stefan Wagner, and Nancy Leveson. 
  https://dspace.mit.edu/handle/1721.1/108230

Introduction to: Systems Theoretic Accident Model & Processes (STAMP) WEBINAR REPLAY (talk)

https://www.youtube.com/watch?v=8bzWvl9OD4 2:33:06


starts at 5:30

notes:

- STAMP: Theoretical Causality Model
- CAST: Accident Analysis
- STPA: Hazard Analysis
- STECA: Early Concept Analysis
- SpecTRM: Model Based System Engineering
- Organizational/Cultural Risk Analysis
- Identifying Leading Indicators
- STPA-Sec: Security Analysis
- 50: STPA example
- Nancy Levenson home page http://sunnyday.mit.edu/

Software Challenges In Achieving Space Safety (paper)

https://dspace.mit.edu/handle/1721.1/58930

The Need for a Paradigm Shift in Safety and Cyber Security (talk)

https://www.youtube.com/watch?v=WBktlCyPl04 1:11:54


An integrated approach to safety and security based on systems theory (paper)

- An integrated approach to safety and security based on systems theory
- William Young and Nancy G. Leveson
- Communications of the ACM, Volume 57 Issue 2, February 2014, Pages 31-35
The SEMA referential framework: Avoiding ambiguities in the terms “security” and “safety” (paper)

- Ludovic Piètre-Cambacédès and Claude Chaudet
- International Journal of Critical Infrastructure Protection
- Volume 3, Issue 2, July 2010, Pages 55-66

Building a safety architecture pattern system (paper)

- Christopher Preschern, Nermin Kajtazovic, and Christian Kreiner
- EuroPLoP ’13 Proceedings of the 18th European Conference on Pattern Languages of Program
- 12 page paper plus 43 page appendix
- https://dl.acm.org/citation.cfm?doid=2739011.2739028

Coping with Defective Software in Medical Devices (paper)

- Steven R. Rakitin
- Computer, Volume 39 Issue 4, April 2006, Page 40-45

Design, Implementation, and Verification of a Reactor Protection System Using HFC6000 (UWO Thesis)

- Michael V. Gverzdys, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/3317/

Predictive Shutdown Systems for Nuclear Power Plants (UWO Thesis)

- Drew J. Rankin, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/4635/

Drivers' needs and safety systems (paper)

- Pierre Van Elslande and Katel Fouquet
- ECCE '08 Proceedings of the 15th European conference on Cognitive ergonomics: the ergonomics of cool interaction, Article No. 8
- http://dl.acm.org/citation.cfm?doid=1473018.1473029

Early Concept Development and Safety Analysis of Future Transportation Systems (paper)

- Cody H. Fleming and Nancy G. Leveson
- IEEE Transactions on Intelligent Transportation Systems, Volume 17 Issue 12, December 2016, Page 3512-3523
Engineering financial safety: a system-theoretic case study from the financial crisis (MIT thesis)

- Spencer, Melissa B. (Melissa Beth)
- https://dspace.mit.edu/handle/1721.1/72903

Engineering safety-related requirements for software-intensive systems (paper)

- Engineering safety-related requirements for software-intensive systems
- Donald Firesmith
- ICSE '06 Proceedings of the 28th international conference on Software engineering, Pages 1047-1048
- http://dl.acm.org/citation.cfm?doid=1134285.1134498

Fail-safe testing of safety-critical systems: a case study and efficiency analysis (paper)

- Fail-safe testing of safety-critical systems: a case study and efficiency analysis
- Ahmed Gario, Anneliese Andrews and Seana Hagerman
- Software Quality Journal, pp 1–46

Inside risks: a risks-related bookshelf (paper)

- Inside risks: a risks-related bookshelf
- Peter G. Neumann
- Communications of the ACM, Volume 39 Issue 4, April 1996, Page 122
- http://dl.acm.org/citation.cfm?doid=227210.247240

Peter Neumann has for many years been publishing articles reporting various failures of computing systems, often in the Communications of the ACM. These can be found in his list of publications http://dl.acm.org/author_page.cfm?id=81100663246&coll=DL&dl=ACM

Integration between requirements engineering and safety analysis: A systematic literature review (paper)

- Integration between requirements engineering and safety analysis: A systematic literature review
- Jéssyka Vilela, Jaelson Castro, Luiz Eduardo G.Martins and Tony Gorschek

Requirements engineering for safety-critical systems: A systematic literature review (paper)

- Requirements engineering for safety-critical systems: A systematic literature review
- Luiz Eduardo G. Martins, Tony Gorschek
- Information and Software Technology, Volume 75, 2016, pp. 71-89

International Conference on Computer Safety, Reliability, and Security (proceedings)
Is safety a subject for science? (paper)

- Erik Hollnagel  
  [https://scholar.google.ca/citations?hl=en&user=X9XNLZMAAAAJ&view_op=list_works&sortby=pubdate]
- Safety Science
- Volume 67, August 2014, Pages 21-24
- [https://www.sciencedirect.com/science/article/pii/S0925753513001756]

Opening the sources of accountability (paper)

- Opening the sources of accountability
- by Shay David
- First Monday, volume 9, number 11 (November 2004),
- [https://firstmonday.org/ojs/index.php/fm/article/view/1185/1105]

Safe operation as a social construct (paper)

- Gene I. Rochlin
- ERGONOMICS, 1999, VOL. 42, NO. 11, 1549 - 1560

Safety-Critical Systems - Professor Martyn Thomas CBE (talk)

[https://www.youtube.com/watch?v=E0igfLciI5k]  57:46

transcript: [https://www.gresham.ac.uk/lectures-and-events/safety-critical-systems]


references IEC 61508 cf [https://en.wikipedia.org/wiki/IEC_61508]

Safety Issues in Advanced AI (talk)

[https://www.youtube.com/watch?v=7gTPZUjvNdE]  33:28

Facing the unknown: the future of humanity (talk)

[https://www.youtube.com/watch?v=h9NB0EQ9tQg]  1:19:30
Success and failure: human as hero -- human as hazard (paper)

- Success and failure: human as hero -- human as hazard
- Carl Sandom
- SCS '07 Proceedings of the twelfth Australian workshop on Safety critical systems and software and safety-related programmable systems - Volume 86, Pages 79-87
- http://dl.acm.org/citation.cfm?id=1387049

Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems (paper)

- Systems-Theoretic Safety Assessment of Robotic Telesurgical Systems
- Homa Alemzadeh, Daniel Chen, Andrew Lewis, Zbigniew Kalbarczyk, Jaishankar Raman, Nancy Leveson and Ravishankar Iyer
- https://link.springer.com/chapter/10.1007%2F978-3-319-24255-2_16

Assessing Dependability with Software Fault Injection: A Survey (paper)

- Roberto Natella, Domenico Cotroneo and Henrique S. Madeira
- ACM Computing Surveys (CSUR)
- https://dl.acm.org/citation.cfm?doid=2856149.2841425

Why do some (weird) people inject faults? (paper)

- João Carreira and João Gabriel Silva
- ACM SIGSOFT Software Engineering Notes
- Volume 23 Issue 1, January 1998
- Pages 42-43
- https://dl.acm.org/citation.cfm?doid=272263.272273

The ethics of safety-critical systems (paper)

- The ethics of safety-critical systems
- Jonathan Bowen
- Communications of the ACM, Volume 43 Issue 4, April 2000, Pages 91-97
- http://dl.acm.org/citation.cfm?doid=332051.332078

ACM Code of Ethics

https://www.acm.org/about-acm/code-of-ethics
http://ethics.acm.org/code-of-ethics
re safety, see item 1.2: Avoid Harm to Others

The Safety of Software – Constructing and Assuring Arguments (U of York thesis)

The Safety of Software – Constructing and Assuring Arguments
Robert Andrew Weaver
PhD Thesis, University of York, Department of Computer Science, September 2003
https://pdfs.semanticscholar.org/9c24/0ec8c2043eb330e7ed4459f3c5c46e61e203.pdf

The unruly power grid (paper)
Notes on Ontario and 2003 North American Power Failure

- 14 August 2003 a power blackout took down North American upper Midwest and Northeast (including London, Ontario)
- "we now know that negligence by a utility in Ohio and lax oversight by a rookie regulator precipitated the blackout that darkened much of the North American upper Midwest and Northeast"
- "according to the U.S. Department of Energy, cost between US $4 billion and $6 billion, was no anomaly"
- "Of course, the very idea of accepting the inevitability of blackouts is utterly rejected by utility officials and politicians. Certainly the mainstream view among power system engineers continues to be that the answer to reliability problems is to make the grids more robust physically, improve simulation techniques and computerized real-time controls, and improve regulation. What the systems theorists suggest is that even if all that is done and done well—as, of course, it should be—the really big outages still will happen more often than they should."
- "Carnegie Mellon team argues that if blackouts are as hard to predict and prevent as tsunamis and earthquakes, we should make it our business to be prepared. They argue that the question is not how to prevent blackouts, but how to survive them."
- recommendations:
  * have wind-up or battery-operated clocks, radios, and flashlights (keep plenty of batteries on hand)
  * keep cans of food for emergencies -- manual can openers work best
  * have cash on hand
  * install an uninterruptible power supply (most only last a few hours)
  * have phone with cord so you can contact people during power outage
- sidenote:
- note: electricity is used to by water pumps in the water distribution system
- initial crash was 24 hours. recovery took longer.
- Aug. 14, 2003: Massive blackout sends Toronto into chaos https://www.youtube.com/watch?v=kkzZVX6ALaA
- Ontario Power Generation shut down many reactors completely. Restarting most took three to four days; the Pickering power plant was back online 29 August (two weeks after initial power failure)
- http://www.lfpress.com/2013/08/14/2003-blackout-10-years-later
- "I remember how the rich part of London, namely Masonville area, got their power back much sooner than everybody else in the city. Those of us living in the armpit of London, - EoA and downtown, we got ours back after about 3 days."
- Essential Items for a Household Emergency Kit (Toronto Hydro) https://www.youtube.com/watch?v=FArYcAUBU9g
- :etonedis

Towards state-of-the-art and future trends in testing of active safety systems (paper)

- Towards state-of-the-art and future trends in testing of active safety systems
- Alessia Knauss, Christian Berger and Henrik Eriksson
- SEsCPS '16 Proceedings of the 2nd International Workshop on Software Engineering for Smart Cyber-Physical Systems, Pages 36-42
- http://dl.acm.org/citation.cfm?doid=2897035.2897037

Verification of safety-critical software (paper)

- Verification of safety-critical software
- B. Scott Andersen and George Romanski
- Communications of the ACM, Volume 54 Issue 10, October 2011, Pages 52-57
- http://dl.acm.org/citation.cfm?doid=2001269.2001286

How accessibility requirements figure into software systems

Accessibility is an important social issue that impacts computer systems that are designed to interact with people. As such, some aspects of it have become codified in laws and standards
At the design end, this topic is often referred to as inclusive design or universal design

- Inclusive Design (talk)
- Making a difference. One app at a time: Disability, Distinction, and Domestic Access (talk)
- Design Meets Disability (talk)
- Haben Girma: “Designing Technology with Accessibility in Mind” | Talks at Google
- Universal Design
- "Making the Digital and Physical Worlds Accessible for People with Disabilities," Eve Andersson
- User-Sensitive Inclusive Design (paper)
- Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
- Ensuring Universal Design in ICT-Solutions - Towards Identifying Critical Success Factors (Thesis)
- Accessibility for Everyone - An Evening with Laura Kalbag (video)
- Entangled With Books: Two Moments In the Evolution of Accessibility and the Notion of Universal Design (thesis)
- User characteristics and the effectiveness of inclusive design for older users of public access systems (Humboldt-Universität zu Berlin thesis)
- 8 80 Cities (Toronto web site)

As always, there is the question of how to approach quantifying how well the design meets the requirements

- An Introduction to the Theory and Measurement of Self Determination (video)

One issue in design of particular interest here is realizing how the system appears to the user – since at the heart of most accessibility issues was that the programmer viewed the system usage differently than the user does (which is not to say that programmers don’t sometimes build systems that they themselves would find hard to use).

- Prejudices, memories, expectations and confidence influence experienced accessibility on the Web (paper)
- Mental Models of Chinese and German Users and Their Implications for MMI: Experiences from the Case Study Navigation System (paper)

One question is how does accessibility related to usability

- Usability as a non-functional requirement
- Accessibility, usability and universal design—positioning and definition of concepts describing person-environment relationships (paper)
- The relationship between accessibility and usability of websites (paper)
- Bridging the gap: between accessibility and usability (paper)

A tie between systems thinking (which relates to many other nonfunctional requirements) and accessibility requirements is the focus of https://wiki.csd.uwo.ca/display/CS96222017/Usability+%2As+%2A+non-functional+requirement

- Systems Thinking in Design: Service Design and self-Services (paper)

While it is important for software to take advantage of the physical properties of the system where it runs, it is also useful to be able to interface with devices that might be added to the default system by a user with special access needs.

- Braille Displays (note)
- Russell Couper: CANUTE, The worlds first working refreshable multiline braille display. (talk)
- Switch Access for Android (talk)
- Telekinetic Control of Android and Other Wonders of the Modern World (talk)
- SpeakYourMind: Aaron typing with his eyes! Powered by EyeTribe and Microsoft (demo)
- CES 2015: The Eye Tribe (interview)
- Tecla Access - Reading an e-book (talk)
- Why the Dvorak keyboard didn’t take over the world (talk)
- the first mouse (demo)

And at the other end of the development process, the question of how do we know that an accessibility requirement is being met is also an issue.

- Auditing Web accessibility: The role of interest organizations in promoting compliance through certification (paper)
- Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities (U Pittsburgh Thesis)
- Paul J. Adam: Web Accessibility Testing Tools and Techniques Paul Bohman (talk)
- Testing for Accessibility Linux.conf.au 2014 -- Perth, Australia (talk)
- The Effectiveness of the Web Accessibility Audit as a Motivational and Educational Tool in Inclusive Web Design (U Dundee thesis)
- Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users (paper)

Some work has been focused on design for specific access issues

- Mobile Gesture-Based User Interfaces for People with Disabilities (talk)
- Iterative Design and Field Trial of an Aphasia-Friendly Email Tool (paper)
- Teach or Design? How Older Adults’ Use of Ticket Vending Machines Could Be More Effective (paper)
- Usability, accessibility and ambient-assisted living: a systematic literature review (paper)
• Exploring the accessibility of banking and finance systems for blind users (paper)
• ACE: A Colour Palette Design Tool for Balancing Aesthetics and Accessibility (paper)
• The Effect of Font Type on Screen Readability by People with Dyslexia (paper)
• Exploring language technologies to provide support to WCAG 2.0 and E2R guidelines (paper)
• Interviews and Observation of Blind Software Developers at Work to Understand Code Navigation Challenges (paper)

Although we usually think of accessibility in terms of human computer interactions, but computers also play an interesting accessibility role in human interactions.

• Rupal Patel: Synthetic Voices as unique as fingerprints (talk)
• Roger Ebert: Remaking My Voice
• Prof. Simon King - Using Speech Synthesis to give Everyone their own Voice (talk)
• “Speech Synthesis,” Kim Silverman (talk)

Much work on accessibility is currently focused on the design of web applications/sites

• Guiding Novice Web Workers in Making Image Descriptions Using Templates (paper)
• Making the web easier to see with opportunistic accessibility improvement (paper)
• Marcy Sutton: 30 Minutes or Less: The Magic of Automated Accessibility Testing | JSConf US 2015 (talk)
• YOW! West 2017 Ann Mwangi - Accessibility: Is it another checkbox to be ticked? (talk)
• The challenges of Web accessibility: The technical and social aspects of a truly universal Web (paper)
• Stanford Seminar - Building an Accessible Web (talk)
• Beyond Web Content Accessibility Guidelines: Expert Accessibility Reviews (paper)
• A Knowledge-Based Framework for Improving Accessibility of Web Sites (paper)
• Measuring website usability for visually impaired people-a modified GOMS analysis (paper)

Google has been involved in some interesting developments in this area, particularly regarding its Android systems

• Google I/O 2011: Creating Accessible Interactive Web Apps using HTML5 (talk)
• Inclusive design and testing: Making your app accessible - Google I/O 2016 (talk)
• Google I/O 2015 - Improve your Android app's accessibility (talk)
• GTAC 2015: Automated Accessibility Testing for Android Applications (talk)

One question is how do existing projects handle accessibility requirements

• Accessibility for Sakai (open source educational software)
• Accessibility of Atlassian Products
• Accessibility Western (web site)
• Accessibility London Ontario Canada (web site)

Some major publications and blogs on accessibility issues (particularly for IT)

• ACM Transactions on Accessible Computing (TACCESS) (journal web site)
• Universal Access in the Information Society: International Journal (journal)
• ASSETS ACM SIGACCESS Conference on Computers and Accessibility (Proceedings web site)
• List of Accessibility Blogs (link) and A11Y Project (resources)
• Gov UK Accessibility blog (blog)
• The Accessibility Hub (blog, Canada)
• Web Axe -- blog and podcast on web accessibility (blog)

Some theses on the topic:

• A Support System for Graphics for Visually Impaired People (UWO Thesis)
• Accessibility of E-Commerce Websites for Vision Impaired Persons (UWO Thesis)
• Examination of the Usability of the IPAD Among Older Adult Consumers (UWO Thesis)
• Redesign of Johar: a framework for developing accessible applications (UWO Thesis)
• Communicating Through a Brain-Computer Interface: Towards the Assessment of Quality of Life in Minimally Conscious and Covertly Aware Vegetative State Patients (UWO Thesis)
• Towards the Development of a Wearable Tremor Suppression Glove (UWO Thesis)
• Development of a Novel Handheld Device for Active Compensation of Physiological Tremor (UWO Thesis)
• Design and Study of Emotions in Virtual Humans for Assistive Technologies (U Waterloo thesis)
• An Assistive Handwashing System with Emotional Intelligence (U Waterloo thesis)

There is a literature on how this topic should be taught

• Best Practices for Teaching Accessibility in University Classrooms: Cultivating Awareness, Understanding, and Appreciation for Diverse Users (paper)
• Principles and Practice of Assistive Technology (MIT OpenCourseware)

8 80 Cities (Toronto web site)

http://www.880cities.org/
Advocating the design of cities that work for both 8 year olds and 80 year olds.

**Accessibility, usability and universal design—positioning and definition of concepts describing person-environment relationships (paper)**

- S. Iwarsson and A. Ståhl
- **Journal** Disability and Rehabilitation
- **Volume** 25, 2003 - Issue 2, pp. 57 - 66
- [http://www.tandfonline.com/doi/abs/10.1080/dre.25.2.57.66](http://www.tandfonline.com/doi/abs/10.1080/dre.25.2.57.66)

**Bridging the gap: between accessibility and usability (paper)**

- Mary Frances Theofanos and Janice (Ginny) Redish
- **Volume** 10 Issue 6, November + December 2003
- **Pages** 36-51
- [https://dl.acm.org/citation.cfm?doid=947226.947227](https://dl.acm.org/citation.cfm?doid=947226.947227)

**The relationship between accessibility and usability of websites (paper)**

- Helen Petrie and Omar Kheir
- **CHI '07 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**
- **Pages** 397-406
- [https://dl.acm.org/citation.cfm?doid=1240624.1240688](https://dl.acm.org/citation.cfm?doid=1240624.1240688)

**Accessibility for Sakai (open source educational software)**

[https://confluence.sakaiproject.org/display/2ACC/Accessibility+Testing+Tools](https://confluence.sakaiproject.org/display/2ACC/Accessibility+Testing+Tools)

**uses:**

- [http://validator.w3.org/](http://validator.w3.org/)
- [http://getfirebug.com/](http://getfirebug.com/)
- [https://www.deque.com/products/fireeyes](https://www.deque.com/products/fireeyes)
- [http://wave.webaim.org/api/](http://wave.webaim.org/api/)
- [http://www.msfw.com/Services/ContrastRatioCalculator](http://www.msfw.com/Services/ContrastRatioCalculator)

**Accessibility London Ontario Canada (web site)**

[https://www.london.ca/city-hall/accessibility/Pages/default.aspx](https://www.london.ca/city-hall/accessibility/Pages/default.aspx)

**relevant accessibility legislation:** [https://www.london.ca/city-hall/accessibility/Pages/Accessibility-Laws.aspx](https://www.london.ca/city-hall/accessibility/Pages/Accessibility-Laws.aspx)
Accessibility of Atlassian Products

https://www.atlassian.com/accessibility

with reference to:

- http://www.section508.gov/  GSA Government-wide Section 508 Accessibility Program (US)
- https://www.w3.org/TR/WCAG20/  Web Content Accessibility Guidelines (WCAG) 2.0 (2008)

details:

- https://confluence.atlassian.com/display/ACCESSIBILITY/Confluence+VPAT+for+Section+508+compliance?_ga=2.237974408.963221070.1501875066-1399075428.1468200723 (Confluence)
- https://confluence.atlassian.com/display/ACCESSIBILITY/Bitbucket+Server+VPAT+for+Section+508+compliance?_ga=2.237974408.963221070.1501875066-1399075428.1468200723 (BitBucket)

related:

- https://www.communardo.de/accessibility-working-atlassian-confluence/ product
- https://clients.collegesource.com/home/display/TFO/Accessibility
- https://terc.edu/display/About/Accessibility+Standards

Accessibility Western (web site)

http://accessibility.uwo.ca/

ACE: A Colour Palette Design Tool for Balancing Aesthetics and Accessibility (paper)

- ACE: A Colour Palette Design Tool for Balancing Aesthetics and Accessibility
- Garreth W. Tigwell, David R. Flalla and Neil D. Archibald
- ACM Transactions on Accessible Computing (TACCESS), Volume 9 Issue 2, January 2017, Article No. 5
- http://dl.acm.org/citation.cfm?doid=3015565.3014588

http://daprlab.com/ace/

ACM Transactions on Accessible Computing (TACCESS) (journal web site)

http://dl.acm.org/pub.cfm?id=J1156&CFID=852668125

See Publication Archive tab for list of back issues. Journal started in 2008 and still active.

A Knowledge-Based Framework for Improving Accessibility of Web Sites (paper)

- Pelzetter J.
- https://link.springer.com/chapter/10.1007/978-3-319-58451-5_17
- see also http://ontologydesignpatterns.org/wiki/Main_Page
An Introduction to the Theory and Measurement of Self Determination (video)

https://www.youtube.com/watch?v=Sz2f5XWOyqw1:03:03


based on:

- Understanding the Construct of Self-Determination
- Karrie A. Shogren, Michael L. Wehmeyer, Susan B. Palmer, Jane Helen Soukup, Todd D. Little, Nancy Garner, Margaret Lawrence
- Assessment for Effective Intervention, v33, no 2, March 2008, 94 -- 107

National Gateway on Self-Determination http://ngsd.org/

ARC – for people with intellectual and developmental disabilities https://www.thearc.org/


AIR Self-Determination Assessments


ASSETS ACM SIGACCESS Conference on Computers and Accessibility (Proceedings web site)

http://dl.acm.org/event.cfm?id=RE205

click Publication Archive tab to access individual proceedings. First proceeding was 1994. In 2016, we reach 18th proceeding.

Of related interest: http://dl.acm.org/citation.cfm?id=2667081 Proceedings of the First International Workshop on Usability and Accessibility Focused Requirements Engineering, ICSE ’12 34th International Conference on Software Engineering, Zurich, Switzerland — June 02 - 09, 2012

A Support System for Graphics for Visually Impaired People (UWO Thesis)

- A Support System for Graphics for Visually Impaired People
- Hao Xu, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1124/

Accessibility of E-Commerce Websites for Vision Impaired Persons (UWO Thesis)

- Accessibility of E-Commerce Websites for Vision Impaired Persons
- Roopa Bose, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1916/

Examination of the Usability of the IPAD Among Older Adult Consumers (UWO Thesis)

- Examination of the Usability of the IPAD Among Older Adult Consumers
- Mike J. Voumvakis, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/2381/

Redesign of Johar: a framework for developing accessible applications (UWO Thesis)

- Redesign of Johar: a framework for developing accessible applications
- Oladapo Oyebode, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1761/
Best Practices for Teaching Accessibility in University Classrooms: Cultivating Awareness, Understanding, and Appreciation for Diverse Users (paper)

- Best Practices for Teaching Accessibility in University Classrooms: Cultivating Awareness, Understanding, and Appreciation for Diverse Users
- Cynthia Putnam, Maria Dahman, Emma Rose, Jinghui Cheng and Glenn Bradford
- ACM Transactions on Accessible Computing (TACCESS), Volume 8 Issue 4, May 2016, Article No. 13
- http://dl.acm.org/citation.cfm?doid=2905046.2831424

Beyond Web Content Accessibility Guidelines: Expert Accessibility Reviews (paper)

- Rocio Calvo, Faezeh Seyedarabi and Andreas Savva
- DSAI 2016 Proceedings of the 7th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion
- Pages 77-84
- https://dl.acm.org/citation.cfm?doid=3019943.3019955

Design and Study of Emotions in Virtual Humans for Assistive Technologies (U Waterloo thesis)

- Malhotra, Aarti
- 2016-04-27
- https://uwspace.uwaterloo.ca/handle/10012/10404

An Assistive Handwashing System with Emotional Intelligence (U Waterloo thesis)

- Lin, Luyuan
- 2014-08-19
- https://uwspace.uwaterloo.ca/handle/10012/8654

Ensuring Universal Design in ICT-Solutions - Towards Identifying Critical Success Factors (Thesis)

- Harder, Susanne Klungland
- Norwegian University of Science and Technology, MS Thesis, June 2017, 160 pages
- https://brage.bibsys.no/xmlui/bitstream/handle/11250/2448937/18056_FULLTEXT.pdf?sequence=1&isAllowed=y

Entangled With Books: Two Moments In the Evolution of Accessibility and the Notion of Universal Design (thesis)

- Hollis Pierce
- Carleton University, HIST 4910 - A (https://calendar.carleton.ca/search/?P=HIST%204910), April 8, 2015, 50 pages

home page: http://hollispeirce.grahamresearchfellow.org/

Evaluation and Enhancement of Web Content Accessibility for Persons with Disabilities (U Pittsburgh Thesis)
Exploring language technologies to provide support to WCAG 2.0 and E2R guidelines (paper)

- Lourdes Moreno, Paloma Martínez, Isabel Segura-Bedmar and Ricardo Revert
- Interacción '15 Proceedings of the XVI International Conference on Human Computer Interaction
- https://dl.acm.org/citation.cfm?doid=2829875.2829927

Guiding Novice Web Workers in Making Image Descriptions Using Templates (paper)

- Guiding Novice Web Workers in Making Image Descriptions Using Templates
- Valerie S. Morash, Yue-Ting Siu, Joshua A. Miele, Lucia Hasty and Steven Landau
- ACM Transactions on Accessible Computing (TACCESS), Volume 7 Issue 4, November 2015, Article No. 12
- http://dl.acm.org/citation.cfm?doid=2847216.2764916

Inclusive Design (talk)

https://www.youtube.com/watch?v=iSkoZULbsTo 46:33

quote: "It is stairs that make a building inaccessible, not the wheelchair."


Design Meets Disability (talk) (book)

https://www.youtube.com/watch?v=uMngJP8pP0c 1:19:37

Design meets disability

by Pullin, Graham:

IN LIBRARY, NK1110.P85 2009, King's Stacks (Second Floor)
IN LIBRARY, WB320.P982d 2009, Taylor Library Stack 4 (S4) - Regular Loan

http://alpha.lib.uwo.ca/record=b4717639

https://mitpress.mit.edu/books/design-meets-disability

https://www.amazon.ca/Design-Meets-Disability-Graham-Pullin/dp/0262516748

Haben Girma: "Designing Technology with Accessibility in Mind" | Talks at Google (talk)

https://www.youtube.com/watch?v=44VEwsBfx0 41:28

Inclusive design of ICT: The challenge of diversity (U Oslo thesis)
Inclusive design of ICT: The challenge of diversity
Kristin Skeide Fuglerud
University of Oslo

Making a difference. One app at a time: Disability, Distinction, and Domestic Access (talk)
https://www.youtube.com/watch?v=abN6sxHmUhk 56:59

Mobile Gesture-Based User Interfaces for People with Disabilities (talk)
https://www.youtube.com/watch?v=4wWE4T8-5Hc 1:11:32

User characteristics and the effectiveness of inclusive design for older users of public access systems (Humboldt-Universität zu Berlin thesis)
User characteristics and the effectiveness of inclusive design for older users of public access systems
Sengpiel, Michael
Humboldt-Universität zu Berlin, Lebenswissenschaftliche Fakultät
PhD Thesis, 2015-03-09
https://edoc.hu-berlin.de/handle/18452/17871

User-Sensitive Inclusive Design (paper)

User-Sensitive Inclusive Design
A. F. Newell, P. Gregor, M. Morgan, G. Pullin, and C. Macaulay
in Universal Access in the Information Society

Interviews and Observation of Blind Software Developers at Work to Understand Code Navigation Challenges (paper)

- Khaled Albusays, Stephanie Ludi and Matt Huenerfauth
- ASSETS ’17 Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility
- Pages 91-100
- https://dl.acm.org/citation.cfm?doid=3132525.3132550

Iterative Design and Field Trial of an Aphasia-Friendly Email Tool (paper)

- Iterative Design and Field Trial of an Aphasia-Friendly Email Tool
- Abdullah Al Mahmud and Jean-Bernard Martens
- ACM Transactions on Accessible Computing (TACCESS), Volume 7 Issue 4, November 2015, Article No. 13
- http://dl.acm.org/citation.cfm?doid=2847216.2790305

List of Accessibility Blogs (link)

https://www.digitala11y.com/accessibility-blogs/

see also: A11Y Project (resources)
A11Y Project (resources)
https://a11yproject.com/
https://a11yproject.com/patterns
https://a11yproject.com/checklist
https://a11yproject.com/resources
https://github.com/a11yproject/a11yproject.com/

Gov UK Accessibility blog (blog)
https://accessibility.blog.gov.uk/

The Accessibility Hub (blog, Canada)
http://www.theaccessibilityhub.ca/blog

Web Axe -- blog and podcast on web accessibility (blog)
http://www.webaxe.org/

Making the web easier to see with opportunistic accessibility improvement (paper)

- Jeffrey P. Bigham
- UIST '14 Proceedings of the 27th annual ACM symposium on User interface software and technology, Pages 117-122
- https://dl.acm.org/citation.cfm?doid=2642918.2647357

Measuring website usability for visually impaired people-a modified GOMS analysis (paper)

- Henrik Tonn-Eichstädt
- Assets '06 Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility
- Pages 55-62
- https://dl.acm.org/citation.cfm?doid=1168987.1168998

Mental Models of Chinese and German Users and Their Implications for MMI: Experiences from the Case Study Navigation System (paper)

- Barbara Knapp
- https://link.springer.com/chapter/10.1007/978-3-540-73105-4_97

Prejudices, memories, expectations and confidence influence experienced accessibility on the Web (paper)

- Amaia Aizpurua, Myriam Arrue and Markel Vigo
- Computers in Human Behavior
- Volume 51 Issue PA, October 2015
- Pages 152-160
Principles and Practice of Assistive Technology (MIT OpenCourseware)

- Instructor(s): William Li, Grace Teo, and Prof. Robert Miller
- MIT Course Number 6.811
- As Taught In Fall 2014
- Level: Undergraduate

Rupal Patel: Synthetic Voices as unique as fingerprints (talk)

https://www.ted.com/talks/rupal_patel_synthetic_voices_as_unique_as_fingerprints 11:34

Many people depend on synthetic voices to communicate with others (cf. Stephen Hawking). This talk raises the issue that not everyone in this group should have the same voice, but rather they should have voices that reflect themselves like the voices of people who can speak naturally.

Some related links on open source speech synthesis systems:
- http://espeak.sourceforge.net/ (written in C)
- http://festvox.org/
- http://www.festvox.org/flite/ (written in C)
- http://www.cstr.ed.ac.uk/projects/festival/ (written in C++)
- http://www.voxforge.org/

"Speech Synthesis," Kim Silverman (talk)

https://www.youtube.com/watch?time_continue=1&v=7mjh0PSUv0M 1:31:29

Prof. Simon King - Using Speech Synthesis to give Everyone their own Voice (talk)

https://www.youtube.com/watch?v=xzL-pxcpo-E 55:32

Roger Ebert: Remaking My Voice


Section 508 Amendment to the Rehabilitation Act of 1973 (US Accessibility Legal Requirement) (links)

https://en.wikipedia.org/wiki/Section_508_Amendment_to_the_Rehabilitation_Act_of_1973
http://webaim.org/standards/508/checklist
https://www.section508.gov/

Systems Thinking in Design: Service Design and self-Services (paper)

- John Darzentas, Jenny Siobhane Darzentas
- Form Akademisk, volume 7, number 4 (2014), 18 pages
Teach or Design? How Older Adults’ Use of Ticket Vending Machines Could Be More Effective (paper)

Sustainable E2 mobility services for elderly people — Platform system architecture (paper)

Usability, accessibility and ambient-assisted living: a systematic literature review (paper)

The challenges of Web accessibility: The technical and social aspects of a truly universal Web (paper)

Auditing Web accessibility: The role of interest organizations in promoting compliance through certification (paper)

Blurred lines: Accessibility, disability, and definitional limitations (paper)

Code in action: Closing the black box of WCAG 2.0, A Latourian reading of Web accessibility (paper)
Disability, human rights, and social justice: The ongoing struggle for online accessibility and equality (paper)

- Disability, human rights, and social justice: The ongoing struggle for online accessibility and equality
- by Paul T. Jaeger.
- First Monday, Volume 20, Number 9 - 7 September 2015

Exploring the accessibility of banking and finance systems for blind users (paper)

- Exploring the accessibility of banking and finance systems for blind users
- by Brian Wentz, Dung (June) Pham, and Kailee Tressler.
- First Monday, Volume 22, Number 3 - 6 March 2017

Stanford Seminar - Building an Accessible Web (talk)

https://www.youtube.com/watch?v=1T30sjYdPPE 1:08:59

The Effectiveness of the Web Accessibility Audit as a Motivational and Educational Tool in Inclusive Web Design (U Dundee thesis)

The Effectiveness of the Web Accessibility Audit as a Motivational and Educational Tool in Inclusive Web Design
David R. Sloan
https://pdfs.semanticscholar.org/0420/21f694a436e969ecf220bf4ac714980c560a.pdf

Web accessibility guidelines for the 2020s (paper)

- Michael Cooper
- W4A '16 Proceedings of the 13th Web for All Conference
- 4 pages.
- https://dl.acm.org/citation.cfm?doid=2899475.2899492

The Effect of Font Type on Screen Readability by People with Dyslexia (paper)

- The Effect of Font Type on Screen Readability by People with Dyslexia
- Luz Rello and Ricardo Baeza-Yates
- ACM Transactions on Accessible Computing (TACCESS), Volume 8 Issue 4, May 2016, Article No. 15
- http://dl.acm.org/citation.cfm?doid=2905046.2897736

Universal Access in the Information Society: International Journal (journal)

Universal Access in the Information Society: International Journal (journal)
https://link.springer.com/journal/10209

Universal Design (talk)

https://www.youtube.com/watch?v=3C62aErzzWI 24:16

Web accessibility: Filtering redundant and irrelevant information improves website usability for blind users (paper)
Web Content Accessibility Guidelines (WCAG) 2.0 (web document)

https://www.w3.org/TR/WCAG20/

Why the Dvorak keyboard didn't take over the world (talk)

https://www.youtube.com/watch?v=ZnUBl90tayI 3:48

the first mouse (demo)

https://www.youtube.com/watch?v=1MPJZ6M52dI 1:54

Note layout: chord keyboard on left, standard keyboard in middle, and mouse on right. For more on chord keyboards see https://en.wikipedia.org/wiki/Chorded_keyboard

For more on the mouse, see https://en.wikipedia.org/wiki/Computer_mouse

YOW! West 2017 Ann Mwangi - Accessibility: Is it another checkbox to be ticked? (talk)

https://www.youtube.com/watch?v=9OosNleWb38 23:08

"Making the Digital and Physical Worlds Accessible for People with Disabilities," Eve Andersson (talk)

https://www.youtube.com/watch?v=gK00-nZESTM 1:03:18

Google I/O 2011: Creating Accessible Interactive Web Apps using HTML5 (talk)

https://www.youtube.com/watch?v=fMrkCoqgoxw 49:37
https://code.google.com/archive/p/google-axs-chrome/
http://accessibleweb.info/tag/chromeshades/

in questions part references: Section 508 Amendment to the Rehabilitation Act of 1973 (US Accessibility Legal Requirement)

Google I/O 2015 - Improve your Android app's accessibility (talk)

https://www.youtube.com/watch?v=euEstfNRSzw4 41:18

Braille Displays (note)
Russell Couper: CANUTE, The world's first working refreshable multiline braille display. (talk)

Switch Access for Android (talk)

GTAC 2015: Automated Accessibility Testing for Android Applications (talk)

Inclusive design and testing: Making your app accessible - Google I/O 2016 (talk)

Marcy Sutton: 30 Minutes or Less: The Magic of Automated Accessibility Testing | JSConf US 2015 (talk)

Paul J. Adam: Web Accessibility Testing Tools and Techniques Paul Bohman (talk)

Telekinetic Control of Android and Other Wonders of the Modern World (talk)

GoogleTechTalks 2011
Speaker from Komodo OpenLabs (Ontario)
Communicating Through a Brain-Computer Interface: Towards the Assessment of Quality of Life in Minimally Conscious and Covertly Aware Vegetative State Patients (UWO Thesis)

- Communicating Through a Brain-Computer Interface: Towards the Assessment of Quality of Life in Minimally Conscious and Covertly Aware Vegetative State Patients
- Jasmine Tung, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/4418/

SpeakYourMind: Aaron typing with his eyes! Powered by EyeTribe and Microsoft (demo)

https://www.youtube.com/watch?v=DSVVOQIDkn8 1:41

CES 2015: The Eye Tribe (interview)

https://www.youtube.com/watch?v=bFF4Q5cxOJU 6:50
https://www.youtube.com/user/TheEyeTribe/videos
https://en.wikipedia.org/wiki/The_Eye_Tribe

Tecla Access - Reading an e-book (talk)

https://www.youtube.com/watch?v=gXI-hUXiKyM 1:26
https://www.youtube.com/channel/UC30ZY_sS0ipeQK4iFgYsoFQ (Komodo OpenLab youtube channel)
https://gettecla.com/

Towards the Development of a Wearable Tremor Suppression Glove (UWO Thesis)

- Towards the Development of a Wearable Tremor Suppression Glove
- Yue Zhou, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/3432/

Development of a Novel Handheld Device for Active Compensation of Physiological Tremor (UWO Thesis)

- Development of a Novel Handheld Device for Active Compensation of Physiological Tremor
- Abhijit Saxena, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/914/

Testing for Accessibility Linux.conf.au 2014 -- Perth, Australia (talk)

https://www.youtube.com/watch?v=rxh6B3ChLc 45:35

How sustainability requirements figure into software systems

The meaning of sustainability relating to software has a number of competing interpretations as it is still relatively new.

- Framing sustainability as a property of software quality (paper)
System requirements engineering can be viewed as managing a negotiation among various stakeholders in the proposed system.

- Who is the advocate?: stakeholders for sustainability (paper)

One interpretation is to use software to manage computer-based infrastructure's environmental impact to make the whole system environmentally sustainable.

- Academic ICT and Green Cyberinfrastructure (talk)
- A systematic literature review for software sustainability measures (paper)
- GreenOracle: estimating software energy consumption with energy measurement corpora (paper)
- Quality Metrics for Sustainability: The Operational Energy Use of Application Software (Utrecht U thesis)

A different notion is that the software system itself should be sustainable. For example, researchers use software as part of the research process and it is very disruptive when that software stops being supported (either because the company that sold it went out of business or because the person who was maintaining it stopped being interested in doing so).

- Cyberinfrastructure Software Sustainability & Reusability Workshop - Software and the Long Haul (talk)
- Creating Sustainable Cyberinfrastructures (talk)
- Cultivating Sustainable Software for Research (talk)
- Cyberinfrastructure Software Sustainability & Reusability Workshop - An Industry View (talk)
- Cyberinfrastructure Software Sustainability & Reusability Workshop - Software Sustainability (talk)
- Cyberinfrastructure Software Sustainability & Reusability Workshop - Sustainable Software (talk)
- Cyberinfrastructure Software Sustainability & Reusability Workshop - Welcome Address (talk)
- Cyberinfrastructure Software Sustainability and Reusability: Report from an NSF-funded workshop (report)

And one can look more closely at the economics of sustaining software systems (the cost of maintenance and upgrades)

- So Much Money for So Little Capability: The Reality of Sustaining DoD Software Systems (slides for talk)
- Measuring software sustainability (paper)
- Sustainability guidelines for long-living software systems (paper)

In particular, that the software can be maintained

- Toward Architecture Knowledge Sustainability: Extending System Longevity (paper)
- Relating Architectural Decay and Sustainability of Software Systems (paper)
- Identifying Architectural Bad Smells (paper)

One can also look at whether or not maintainable software is consistent with sustainable software (can you have both?)

- Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper)

Also one can think about how system change plays out as organizations change over time

- The Management and Sustainability of Organizational Change in Primary Care Adoption of Electronic Medical Record Systems (McMasters thesis)

And, there is, of course, the question of how do we make software sustainable if that is our intent

- Improving Software Sustainability through Data-driven Technical Debt Management (talk)
- Software Engineering Aspects of Green and Sustainable Software: A Systematic Mapping Study (paper)
- Sustainability design in requirements engineering: state of practice (paper)
- Sustainable software design (paper)
- Systematic mapping study on software engineering for sustainability (SE4S) (paper)
- Three Essays on Sustainable Operations Management (UWO Thesis)

It is also important to remember that sustainability isn't just a technical problem, but we also must consider how the human uses the technology

- Lynn Bartram - Human-Centred Systems for Sustainable Living (talk)
Patricia Lago: Sustainable Software for a Digital Society (talk)
Performativity in sustainable interaction: the case of seasonal grocery shopping in ecofriends (paper)

Proceedings in the area of software sustainability

- International Workshop on Green and Sustainable Software (Proceedings web sites)
- LiMITS 15’16’17 – Workshop on Computing Within Limits (proceedings)
- Journal of Open Research Software (journal)

And, of course, there is a literature on how to teach and research software sustainability

- Jumpstart sustainability in seminars: hands-on experiences in class (paper)
- Infusing sustainability into software engineering education: Lessons learned from capstone projects (paper)
- Preparing research projects for sustainable software engineering in society (paper)
- A Curriculum on Sustainable Information Communication Technology (paper)
- An international Master's program in green ICT as a contribution to sustainable development (paper)
- Educational aspects of sustainable development analysis: computational models and software (paper)

A major historical marker in the environmental movement view of sustainability was a systems based analysis of world population growth and resource usage from the Club of Rome called Limits to Growth https://en.wikipedia.org/wiki/The_Limits_to_Growth in 1972. It can be viewed as predicting the Climate Change problems that we are having today.

- The Club of Rome and Limits To Growth: Achieving the Best Possible Future (talk)
- Aggregating Evidence in Climate Science: Consilience, Robustness and the Wisdom of Multiple Models (UWO Thesis)
- A System Dynamics Based Integrated Assessment Modelling of Global-Regional Climate Change: A Model for Analyzing the Behaviour of the Social-Energy-Economy-Climate System (UWO Thesis)
- A Systems Approach to Modelling the Effects of Climate Change on Agroforestry: A Case Study in Western Tanzania (UWO Thesis)
- Biopolitics of Climate Change: Carbon Commodities, Environmental Profanations, and the Lost Innocence of Use-Value (UWO Thesis)
- On the Economics of Climate Change and its Effects (UWO Thesis)

While current system problems are large, it is worthwhile putting them in the context of past problems (which we survived, or at least delayed to become our current problems)

- Thomas HAGER - Haber-Bosch: The Discovery that changed the World (talk)
- Keynote Address: Mr. Thomas Hager (talk)

Academic ICT and Green Cyberinfrastructure (talk)

https://www.youtube.com/watch?v=u8_2KCNfPbU 1:00:30

Analyzing the Harmful Effect of God Class Refactoring on Power Consumption (paper)

- Analyzing the Harmful Effect of God Class Refactoring on Power Consumption
- Ricardo Perez-Castillo and Mario Piattini

Toward Architecture Knowledge Sustainability: Extending System Longevity (paper)

- Rafael Capilla ; Elisa Yumi Nakagawa ; Uwe Zdun ; Carlos Carrillo

Identifying Architectural Bad Smells (paper)

- Joshua Garcia ; Daniel Popescu ; George Edwards ; Nenad Medvidovic

Relating Architectural Decay and Sustainability of Software Systems (paper)
A systematic literature review for software sustainability measures (paper)

- A systematic literature review for software sustainability measures
- Coral Calero, Manuel F. Bertoa and Ma Ángeles Moraga
- http://dl.acm.org/citation.cfm?id=2662702

Cyberinfrastructure Software Sustainability & Reusability Workshop - Software and the Long Haul (talk)

https://www.youtube.com/watch?v=pChPpqrYtmw  51:46

Creating Sustainable Cyberinfrastructures (talk)

https://www.youtube.com/watch?v=US2Mb32qD4c  14:34

Cultivating Sustainable Software for Research (talk)

https://www.youtube.com/watch?v=fLpdb_Vfl2Q  1:03:55

Cyberinfrastructure Software Sustainability & Reusability Workshop - An Industry View (talk)

https://www.youtube.com/watch?v=h2g31h1qV_0  1:00:01

Cyberinfrastructure Software Sustainability & Reusability Workshop - Software Sustainability (talk)

https://www.youtube.com/watch?v=DWrXw1X_3dY  36:06

Cyberinfrastructure Software Sustainability & Reusability Workshop - Sustainable Software (talk)

https://www.youtube.com/watch?v=YUyaLotcZY8  45:38
Cyberinfrastructure Software Sustainability & Reusability Workshop - Welcome Address (talk)

https://www.youtube.com/watch?v=_CxVCCcocyc  21:32

Cyberinfrastructure Software Sustainability and Reusability: Report from an NSF-funded workshop (report)


Framing sustainability as a property of software quality (paper)

- Framing sustainability as a property of software quality
- Patricia Lago, Sedef Akinli Koçak, Ivica Crnkovic and Birgit Penzenstadler
- Communications of the ACM, Volume 58 Issue 10, October 2015, Pages 70-78
- http://dl.acm.org/citation.cfm?doid=2830674.2714560

GreenOracle: estimating software energy consumption with energy measurement corpora (paper)

- Shaiful Alam Chowdhury and Abram Hindle
- MSR '16 Proceedings of the 13th International Conference on Mining Software Repositories
- Pages 49-60
- https://dl.acm.org/citation.cfm?doid=2901739.2901763

Improving Software Sustainability through Data-driven Technical Debt Management (talk)

https://www.youtube.com/watch?v=PXN1k6c_2p0  19:09

Information systems for the age of consequences (paper)

- Information systems for the age of consequences
- by M. Six Silberman.
- First Monday, Volume 20, Number 8 - 3 August 2015

Abstraction, indirection, and Sevareid’s Law: Towards benign computing (paper)

- Abstraction, indirection, and Sevareid’s Law: Towards benign computing
- by Barath Raghavan.
- First Monday, Volume 20, Number 8 - 3 August 2015

Collapse (& other futures) software engineering (paper)

- Collapse (& other futures) software engineering
- by Birgit Penzenstadler, Ankita Raturi, Debra J. Richardson, M. Six Silberman, and Bill Tomlinson.
- First Monday, Volume 20, Number 8 - 3 August 2015
Computing within limits and ICTD (paper)

- Computing within limits and ICTD
- by Jay Chen.
- First Monday, Volume 20, Number 8 - 3 August 2015

Limits and sustainable interaction design: Obsolescence in a future of collapse and resource scarcity (paper)

- Limits and sustainable interaction design: Obsolescence in a future of collapse and resource scarcity
- by Christian Remy and Elaine M. Huang.
- First Monday, Volume 20, Number 8 - 3 August 2015

Preliminary thoughts on a taxonomy of value for sustainable computing (paper)

- Preliminary thoughts on a taxonomy of value for sustainable computing
- by Kentaro Toyama.
- First Monday, Volume 20, Number 8 - 3 August 2015
- http://journals.uic.edu/ojs/index.php/fm/article/view/6122/4841

International Workshop on Green and Sustainable Software (Proceedings websites)

- click Table of Contents tab at web site to access individual articles.

  - Proceedings of the First International Workshop on Green and Sustainable Software
    - Conference Chairs: Rick Kazman, Patricia Lago, Niklaus Meyer, Maurizio Morisio, Hausi A. Müller, Frances Paulisch, Giuseppe Scanniello, and Olaf Zimmermann
    - ICSE '12 34th International Conference on Software Engineering, Zurich, Switzerland — June 02 - 09, 2012, http://dl.acm.org/citation.cfm?id=2663779

  - Proceedings of the 2nd International Workshop on Green and Sustainable Software
    - Conference Chairs: Patricia Lago, Niklaus Meyer, Maurizio Morisio, Hausi A. Müller and Giuseppe Scanniello

  - Proceedings of the 3rd International Workshop on Green and Sustainable Software
    - General Chair: Hausi A. Müller (Program Chairs: Patricia Lago, Maurizio Morisio, Niklaus Meyer and Giuseppe Scanniello)

  - Proceedings of the Fourth International Workshop on Green and Sustainable Software
    - Program Chairs: Patricia Lago and Hausi A. Müller

  - Proceedings of the 5th International Workshop on Green and Sustainable Software
    - ICSE '16 38th International Conference on Software Engineering, Austin, TX, USA — May 14 - 22, 2016, http://dl.acm.org/citation.cfm?id=2896967

Journal of Open Research Software (journal)

Journal of Open Research Software
https://openresearchsoftware.metajnl.com/
Jumpstart sustainability in seminars: hands-on experiences in class (paper)

- Jumpstart sustainability in seminars: hands-on experiences in class
- Birgit Penzenstadler and Veronika Bauer
- CSERC '12 Proceedings of Second Computer Science Education Research Conference, Pages 37-44.
- http://dl.acm.org/citation.cfm?doid=2421277.2421282

A Curriculum on Sustainable Information Communication Technology (paper)

A Curriculum on Sustainable Information Communication Technology (July 31, 2015).
Özkan, Bar and Mishra, Alok,
https://ssrn.com/abstract=2660765

An international Master's program in green ICT as a contribution to sustainable development (paper)

An international Master’s program in green ICT as a contribution to sustainable development
Alexandra Klimova, Eric Rondeau, Karl Andersson, Jari Porras, Adrei Rybin, and Arkady Zaslavsky
Journal of Cleaner Production
Volume 135, 1 November 2016, Pages 223-239

Educational aspects of sustainable development analysis: computational models and software (paper)

Educational aspects of sustainable development analysis: computational models and software
Anatoly Kurkovsky
Journal of Computing Sciences in Colleges archive
Volume 21 Issue 4, April 2006, Pages 24-31
http://dl.acm.org/citation.cfm?id=1127396

Infusing sustainability into software engineering education: Lessons learned from capstone projects (paper)

Infusing sustainability into software engineering education: Lessons learned from capstone projects
Maria Victoria Palacin-Silva, Ahmed Seffah, and Jari Porras
Journal of Cleaner Production
Available online 9 June 2017

Lynn Bartram - Human-Centred Systems for Sustainable Living (talk)

https://www.youtube.com/watch?v=eygwpflvk3M8 1:05:50

slides from earlier version of talk: https://www.slideshare.net/johnnyrodgers/supporting-sustainable-living-aware-homes-and-smart-occupants
Patricia Lago: Sustainable Software for a Digital Society (talk)

https://www.youtube.com/watch?v=3yDbwh8-Zj0  28:23

Performativity in sustainable interaction: the case of seasonal grocery shopping in ecofriends (paper)

• Maria Normark and Jakob Tholander
• https://dl.acm.org/citation.cfm?doid=2556288.2557318

Preparing research projects for sustainable software engineering in society (paper)

• Preparing research projects for sustainable software engineering in society
• Dominik Renzel, István Koren, Ralf Klamma and Matthias Jarke
• ICSE-SEIS ’17 Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Society Track, Pages 23-32
• http://dl.acm.org/citation.cfm?id=3103222

Quality Metrics for Sustainability: The Operational Energy Use of Application Software (Utrecht U thesis)

Quality Metrics for Sustainability: The Operational Energy Use of Application Software
Spauwen, R.A.
Utrecht University, Master thesis, 2015
https://dspace.library.uu.nl/handle/1874/312453

Requirements: Sustainability and Requirements: A Manifesto
ieeeComputerSociety (talk)

https://www.youtube.com/watch?v=PXhFgswJPco 10:18

see also: http://sustainabilitydesign.org/ for copy of manifesto


Sustainability Design and Software: The Karlskrona Manifesto (paper)

Sustainability Design and Software: The Karlskrona Manifesto
Christoph Becker, Ruzanna Chitchyan, Leticia Duboc, Steve Easterbrook, Birgit Penzenstadler, Norbert Seyff, and Colin C. Venters
Software Engineering (ICSE), 2015 IEEE/ACM 37th IEEE International Conference on


• Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century
• Birgit Penzenstadler ; Ankita Raturi ; Debra Richardson ; Bill Tomlinson
• IEEE Software ( Volume: 31, Issue: 3, May-June 2014 ), Page(s): 40 - 47
Characterising sustainability requirements: a new species, red herring, or just an odd fish? (paper)

- Characterising sustainability requirements: a new species, red herring, or just an odd fish?
- Colin C. Venters, Norbert Seyff, Christoph Becker, Stefanie Betz, Ruzanna Chitchyan, Leticia Duboc, Dan McIntyre and Birgit Penzenstadler
- ICSE-SEIS ’17 Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Society Track, Pages 3-12
- http://dl.acm.org/citation.cfm?id=3103220

Software Engineering Aspects of Green and Sustainable Software: A Systematic Mapping Study (paper)

- Software Engineering Aspects of Green and Sustainable Software: A Systematic Mapping Study
- C. Marimuthu and K. Chandrasekaran
- ISEC ’17 Proceedings of the 10th Innovations in Software Engineering Conference, Pages 34-44
- http://dl.acm.org/citation.cfm?doid=3021460.3021464

Software Engineering for Sustainability (thesis)

- Dr. Birgit Penzenstadler
- Habilitationsschrift, 28 Jan 2015, 296 pages.
- TECHNISCHE UNIVERSITÄT MÜNCHEN FAKULTÄT FÜR INFORMATIK

home page: http://birgit.penzenstadler.de/

So Much Money for So Little Capability: The Reality of Sustaining DoD Software Systems (slides for talk)

- So Much Money for So Little Capability: The Reality of Sustaining DoD Software Systems
- David Schneider, Fred Schenker and Grady Campbell
- Software Solutions Symposium 2017
- https://resources.sei.cmu.edu/asset_files/Presentation/2017_017_001_496077.pdf

Measuring software sustainability (paper)

Measuring software sustainability

Sustainability guidelines for long-living software systems (paper)

Sustainability guidelines for long-living software systems
Z. Durdik, B. Klatt, H. Koziolek, K. Krogmann, J. Stammel, and R. Weiss
Software Maintenance (ICSM), 2012 28th IEEE International Conference on


- J. Stammel, Z. Durdik, K. Krogmann, R. Weiss, and H. Koziolek
- Karlsruhe, Germany, Karlsruhe Reports in Informatics 2011 - 2, 2011
- 137 pages.
Sustainability debt: a portfolio-based approach for evaluating sustainability requirements in architectures (paper)

- Sustainability debt: a portfolio-based approach for evaluating sustainability requirements in architectures
- Bendra Ojameruaye, Rami Bahsoon and Leticia Duboc
- ICSE ‘16 Proceedings of the 38th International Conference on Software Engineering Companion, Pages 543-552
- http://dl.acm.org/citation.cfm?doid=2889160.2889218

Sustainability design in requirements engineering: state of practice (paper)

- Sustainability design in requirements engineering: state of practice
- Ruzanna Chitchyan, Christoph Becker, Stefanie Betz, Leticia Duboc, Birgit Penzenstadler, Norbert Seyff and Colin C. Venters
- ICSE ‘16 Proceedings of the 38th International Conference on Software Engineering Companion, Pages 533-542
- http://dl.acm.org/citation.cfm?doid=2889160.2889217

Sustainable software design (paper)

- Sustainable software design
- Martin P. Robillard
- http://dl.acm.org/citation.cfm?doid=2950290.2983983

Systematic mapping study on software engineering for sustainability (SE4S) (paper)

- Systematic mapping study on software engineering for sustainability (SE4S)
- Birgit Penzenstadler, Ankita Raturi, Debra Richardson, Coral Calero, Henning Femmer and Xavier Franch
- EASE ’14 Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, Article No. 14
- http://dl.acm.org/citation.cfm?doid=2601248.2601256

The Club of Rome and Limits To Growth: Achieving the Best Possible Future (talk)

https://www.youtube.com/watch?v=Pc3SWj-hjTE 1:32:33

Aggregating Evidence in Climate Science: Consilience, Robustness and the Wisdom of Multiple Models (UWO Thesis)

- Aggregating Evidence in Climate Science: Consilience, Robustness and the Wisdom of Multiple Models
- Martin A. Vezér, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/2837/

A System Dynamics Based Integrated Assessment Modelling of Global-Regional Climate Change: A Model for Analyzing the Behaviour of the Social-Energy-Economy-Climate System (UWO Thesis)

- A System Dynamics Based Integrated Assessment Modelling of Global-Regional Climate Change: A Model for Analyzing the Behaviour of the Social-Energy-Economy-Climate System
- Mohammad Khaled Akhtar, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/331/
A Systems Approach to Modelling the Effects of Climate Change on Agroforestry: A Case Study in Western Tanzania (UWO Thesis)

- A Systems Approach to Modelling the Effects of Climate Change on Agroforestry: A Case Study in Western Tanzania
  - Elaine M. Samuel, The University of Western Ontario
  - http://ir.lib.uwo.ca/etd/3782/

Biopolitics of Climate Change: Carbon Commodities, Environmental Profanations, and the Lost Innocence of Use-Value (UWO Thesis)

- Biopolitics of Climate Change: Carbon Commodities, Environmental Profanations, and the Lost Innocence of Use-Value
  - Emanuele Leonardi, The University of Western Ontario
  - http://ir.lib.uwo.ca/etd/858/

On the Economics of Climate Change and its Effects (UWO Thesis)

- On the Economics of Climate Change and its Effects
  - Aaron B. Gertz, The University of Western Ontario
  - http://ir.lib.uwo.ca/etd/2980/

Thomas HAGER - Haber-Bosch: The Discovery that changed the World (talk)

https://www.youtube.com/watch?v=zysSnH7GZCE  49:21

Keynote Address: Mr. Thomas Hager (talk)

https://www.youtube.com/watch?v=TvcODpxK8OI  42:06

The Management and Sustainability of Organizational Change in Primary Care Adoption of Electronic Medical Record Systems (McMasters thesis)

The Management and Sustainability of Organizational Change in Primary Care Adoption of Electronic Medical Record Systems
Kerollos, Joseph
https://macsphere.mcmaster.ca/handle/11375/11972

Three Essays on Sustainable Operations Management (UWO Thesis)

- Three Essays on Sustainable Operations Management
  - Sara Hajmohammad, The University of Western Ontario
  - http://ir.lib.uwo.ca/etd/2908/

Towards a definition of sustainability in and for software engineering (paper)

- Towards a definition of sustainability in and for software engineering
  - Birgit Penzenstadler
  - SAC ’13 Proceedings of the 28th Annual ACM Symposium on Applied Computing, Pages 1183-1185
  - http://dl.acm.org/citation.cfm?doid=2480362.2480585
How other non-functional requirements figure into software systems

Functional requirements have a crisp mathematical flavor to them. Non-functional requirements are generally messier. The boundaries between the different types of non-functional requirements can blur. Work done with one type of non-functional requirement can provide insight into possibilities for another one (which is why the course is interesting in comparing and contrasting three different types (safety, accessibility, and sustainability) rather than just investigating one type. Some types of non-functional requirements that are closely related to the ones we are studying are:

- Privacy as a non-functional software system requirement
- Reliability as a non-functional system requirement
- Security as a non-functional software system requirement
- Usability as a non-functional requirement
- Learnability as a non-functional requirement
- Adaptability as a non-functional requirement
- Technology Acceptance as a non-functional requirement

Adaptability as a non-functional requirement

For an overview:

- Personalized adaptation in pervasive systems via non-functional requirements (paper)
- User modelling for adaptive computer systems: a survey of recent developments (paper)

Adaptability is often part of systems that make ‘personalized’ recommendations to a user


One interesting application for adaptability is automated tutors (a version of e-learning)

- The influence of student characteristics on the use of adaptive e-learning material (paper)
- Designing an adaptive web-based learning system based on students’ cognitive styles identified online (paper)
- Proteus: A Lecturer-Friendly Adaptive Tutoring System (paper)
- On design-oriented research and digital learning materials in higher education (Wageningen PhD Thesis)
- Bioprocess-Engineering Education with Web Technology (U of Wageningen PhD Thesis)

On design-oriented research and digital learning materials in higher education (Wageningen PhD Thesis)

- RJM Hartog, PhD thesis
- Wageningen University, 2012
- http://library.wur.nl/WebQuery/wda/2012885
- http://library.wur.nl/WebQuery/wurpubs/fulltext/242709

Personalized adaptation in pervasive systems via non-functional requirements (paper)

- Serral, E., Sernani, P. & Dalpiaz, F. J
- https://doi.org/10.1007/s12652-017-0611-4


- Pete Sawyer ; Nelly Bencomo ; Jon Whittle ; Emmanuel Letier ; Anthony Finkelstein
- Requirements Engineering Conference (RE), 2010 18th IEEE International
The influence of student characteristics on the use of adaptive e-learning material (paper)

- J.R.van Seters, M.A.Ossevoort, J.Tramper, and M.J.Goedhart
- Computers & Education
- Volume 58, Issue 3, April 2012, Pages 942-952

Bioprocess-Engineering Education with Web Technology (U of Wageningen PhD Thesis)

- Sessink, O.
- University of Wageningen, 135p., 2006
- http://library.wur.nl/WebQuery/wurpubs/346294

Designing an adaptive web-based learning system based on students’ cognitive styles identified online (paper)

- Jia-Jiunn Lo, Ya-Chen Chan, Shiou-Wen Yeh

Proteus: A Lecturer-Friendly Adaptive Tutoring System (paper)

- Sessink, Olivier D T; Beetink, Hendrik H; Tramper, Johannes; Hartog, Rob J M
- https://search.proquest.com/docview/211229627?pq-origsite=gscholar


- Joeran Beel
- PhD Thesis, Otto-von-Guericke University Magdeburg, Germany

User modelling for adaptive computer systems: a survey of recent developments (paper)

- Michael F. McTear
- Artificial Intelligence Review

Learnability as a non-functional requirement

Overview of learnability issues

- A survey of software learnability: metrics, methodologies and guidelines (paper)

A survey of software learnability: metrics, methodologies and guidelines (paper)

- Tovi Grossman, George Fitzmaurice and Ramtin Attar
- CHI '09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems
- Pages 649-658
Privacy as a non-functional software system requirement

A thesis done at Western related to privacy is

- Privacy in Cooperative Distributed Systems: Modeling and Protection Framework (UWO Thesis)
- Protecting Personal Private Information in Collaborative Environments (UWO Thesis)

Privacy as a legal issue

- The Internet of Things: Implications for Consumer Privacy under Canadian Law (paper)
- Privacy Impacts of IoT Devices: A SmartTV Case Study (paper)
- Analyzing Regulatory Rules for Privacy and Security Requirements (paper)

Relevant governmental organizations

- Office of the Privacy Commissioner of Canada (web site)
- Information and Privacy Commissioner of Ontario (web site)

Blogs:

- The Ad Contrarian Blog

Office of the Privacy Commissioner of Canada (web site)


Information and Privacy Commissioner of Ontario (web site)

https://www.ipc.on.ca/?redirect=https://www.ipc.on.ca/

Time to modernize tools to protect personal information, Privacy Commissioner urges in annual report (news release)


refs:


- Privacy in Cooperative Distributed Systems: Modeling and Protection Framework
- Afshan Samani, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/2777/

Protecting Personal Private Information in Collaborative Environments (UWO Thesis)

- Protecting Personal Private Information in Collaborative Environments
- David S. Allison, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/2105/
Reliability as a non-functional system requirement

Failure handling in distributed systems

- Erlang (paper)
- Making reliable distributed systems in the presence of software errors (thesis) by Joe Armstrong, developer of Erlang

One can also approach this as building ‘fault tolerant systems’

- Evolution of fault tolerance (talk)
- Designing Fault Tolerant Applications (Amazon webinar)
- Ruby Conf 2013 - Fault Tolerant Data: Surviving the Zombie Apocalypse (talk)
- LambdaConf 2015 - Fault Tolerance on the Cheap Making Systems That Probably Won't Fall Over (talk)

Conference:

- Software Reliability Engineering (ISSRE), International Symposium on (proceedings)

A thesis done at Western related to reliability is

- Reliability Models Applied to Smartphone Applications (UWO Thesis)

Erlang (paper)

- Joe Armstrong
  - Communications of the ACM
  - Volume 53 Issue 9, September 2010
  - Pages 68-75
  - https://dl.acm.org/citation.cfm?doid=1810891.1810910

Making reliable distributed systems in the presence of software errors (thesis)

- Joe Armstrong
  - PhD Thesis, Royal Institute of Technology, Stockholm, Sweden
  - December 2003

Evolution of fault tolerance (talk)

https://www.youtube.com/watch?v=4tN_mJcMOYI 31:23
Designing Fault Tolerant Applications (Amazon webinar)

https://www.youtube.com/watch?v=9BrmHoyFJUY 57:07
LambdaConf 2015 - Fault Tolerance on the Cheap Making Systems That Probably Won't Fall Over (talk)

https://www.youtube.com/watch?v=2VYsLusnT3Q 47:45
Ruby Conf 2013 - Fault Tolerant Data: Surviving the Zombie Apocalypse (talk)
Security as a non-functional software system requirement

Useful blogs tracking issues related to security

- Schneier on Security (blog)

As well as software security often becoming a news item

- New Cyber Frontiers (video)

Some introductions to security

- Computer security in the real world (paper)
- Cyber Threats and Cyber Security (talk)
- What Every Engineer Needs to Know About Security and Where to Learn It (talk)
- Maroochy water breach (talk) case study presented by Ian Sommerville
- Software and the mundane management of air travel (paper)
- Why information security is hard - an economic perspective (paper)
- "Building Secure Cultures" by Leigh Honeywell (talk)
- "Incident insights from NASA, NTSB, and the CDC" by Emil Stolarsky (talk)

As with safety, it is important to view security programs/algorithms in the context where they are executed and, in particular, how they interact with humans in such systems.

- Why cryptosystems fail (paper)
- EMV: why payment systems fail (paper)
- Measuring the security impacts of password policies using cognitive behavioral agent-based modeling (paper)
- Cryptographic voting protocols: a systems perspective (paper)
- Cryptanalytic Attacks on Pseudorandom Number Generators (paper)
- Designing security into software (MIT thesis)
- Using STAMP to understand recent increases in malicious software activity (MIT thesis)
- Applying systems thinking to healthcare data cybersecurity (MIT thesis)

As with all non-functional requirements, having a way to verify that they have been met raises interesting issues

- Software security testing (paper)
- Security Testing Fundamentals - Ken van Wyk (talk)
- An empirical study on the effectiveness of security code review (paper)
- Why security testing is hard (paper)
- Software penetration testing (paper)
- One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques (paper)
- Model Checking An Entire Linux Distribution for Security Violations (paper)
- Software Vulnerability Discovery Techniques: A Survey (paper)
- VCCFinder: Finding Potential Vulnerabilities in Open-Source Projects to Assist Code Audits (paper)
- An overview of vulnerability assessment and penetration testing techniques (paper)

Of course, before you can verify, you need to figure out what is being required in the first place

- Analyzing Regulatory Rules for Privacy and Security Requirements (paper)
- Identifying vulnerabilities and critical requirements using criminal court proceedings (paper)
- Legally "reasonable" security requirements: A 10-year FTC retrospective (paper)
- Privacy Impacts of IoT Devices: A SmartTV Case Study (paper)
- The Internet of Things: Implications for Consumer Privacy under Canadian Law (paper)
- A survey of compliance issues in cloud computing (paper)
• General Requirements of a Hybrid-Modeling Framework for Cyber Security (paper)

Then there is the question of how to implement the requirements
• Selecting Security Patterns that Fulfill Security Requirements (paper)

And, of course, how to fix things when there is a problem
• Mitigating program security vulnerabilities: Approaches and challenges (paper)

Since computer networks go out of their way to be accessible to attackers (wireless being more vulnerable than wired), much of the study of security of computer systems focuses on issues relating to networking
• Comprehensive study of web application attacks and classification (paper)
• Surviving the Web: A Journey into Web Session Security (paper)
• Formal methods for web security (paper -- survey)
• Survey on JavaScript security policies and their enforcement mechanisms in a web browser (paper)
• Timing Analysis of Keystrokes and Timing Attacks on SSH (paper)
• Practical Keystroke Timing Attacks in Sandboxed JavaScript. (paper)
• Security collapse in the HTTPS market (paper)
• The web won't be safe or secure until we break it (paper)
• The most dangerous code in the world: validating SSL certificates in non-browser software (paper)
• How Internet Resources Might Be Helping You Develop Faster but Less Securely (paper)
• Software-defined Networking enabled Resource Management and Security Provisioning in 5G Heterogeneous Networks (UWO Thesis)
• Secure OFDM System Design for Wireless Communications Hao Li, The University of Western Ontario (UWO Thesis)
• Security Protocol Suite for Preventing Cloud-based Denial-of-Service Attacks (UWO Thesis)
• Risk Assessment for IoT : a system evaluation of the smart home and its cybersecurity imperative (MIT thesis)
• A systems perspective on cybersecurity in the cloud : frameworks, metrics and migration strategy (MIT thesis)

On a smaller scale than networks, databases often brings together information managed by different security policies becoming another focus for security study
• An Access Control Model for NoSQL Databases (UWO Thesis)

Actually even ram memory is a problem
• One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation (paper)

On the plus side, there are some interesting technical discoveries regarding security techniques
• Verifying computations without reexecuting them (paper)

Journals related to security include:
• ACM Transactions on Privacy and Security (TOPS) (Journal web site)

Books related to security include:
• Building secure software: how to avoid security problems the right way (book)
• Security engineering: a guide to building dependable distributed systems (book)

A look at security from the view of system administrators
• How Do System Administrators Resolve Access-Denied Issues in the Real World? (paper)
• Information Security at Western (Web Site)

"Building Secure Cultures" by Leigh Honeywell (talk)

https://www.youtube.com/watch?v=2BvVZU4IPKc 26:16
13:00 minimal viable software development lifecycle for security
16:20 discussion of checklists

"Incident insights from NASA, NTSB, and the CDC" by Emil Stolarsky (talk)

https://www.youtube.com/watch?v=ODYO2MPymJ4 35:38
Analyzing Regulatory Rules for Privacy and Security Requirements (paper)

- Travis Breaux and Annie Antón
- Page(s): 5 - 20

A survey of compliance issues in cloud computing (paper)

- Dereje Yimam and Eduardo B. Fernandez
- Journal of Internet Services and Applications
- December 2016, 7:5, 12 pages.

Identifying vulnerabilities and critical requirements using criminal court proceedings (paper)

- Travis D. Breaux, Jonathan D. Lewis, Paul N. Otto and Annie I. Antón
- SAC ’09 Proceedings of the 2009 ACM symposium on Applied Computing
- Pages 355-359
- [https://dl.acm.org/citation.cfm?doid=1529282.1529360](https://dl.acm.org/citation.cfm?doid=1529282.1529360)

Legally "reasonable" security requirements: A 10-year FTC retrospective (paper)

- Travis D. Breaux and David L. Baumer
- Computers and Security
- Volume 30 Issue 4, June, 2011
- Pages 178-193

An empirical study on the effectiveness of security code review (paper)

- Anne Edmundson, Brian Holtkamp, Emanuel Rivera, Matthew Finifter, Adrian Mettler and David Wagner
- ESSoS’13 Proceedings of the 5th international conference on Engineering Secure Software and Systems
- Pages 197-212
- [https://link.springer.com/chapter/10.1007%2F978-3-642-36563-8_14](https://link.springer.com/chapter/10.1007%2F978-3-642-36563-8_14)

Applying systems thinking to healthcare data cybersecurity (MIT thesis)

Chung, Kristie (Kristie J.)
Thesis: S.M. in Engineering and Management, Massachusetts Institute of Technology, Engineering Systems Division, 2015
[https://dspace.mit.edu/handle/1721.1/105307](https://dspace.mit.edu/handle/1721.1/105307)

A systems perspective on cybersecurity in the cloud: frameworks, metrics and migration strategy (MIT thesis)
Raina, Ravi  
https://dspace.mit.edu/handle/1721.1/107602

Building secure software: how to avoid security problems the right way (book)

- Building secure software: how to avoid security problems the right way
- by Viega, John; McGraw, Gary
- Addison-Wesley professional computing series, 2002
- Book:
  - QA76.76.D47V857 2002, Taylor Library Stack 6 (S6) - Regular Loan
  - http://alpha.lib.uwo.ca/record=b3129436

Computer security in the real world (paper)

- Computer security in the real world
- B.W. Lampson
- Computer ( Volume: 37, Issue: 6, June 2004 ), Page(s): 37 - 46.

Cryptanalytic Attacks on Pseudorandom Number Generators (paper)

- John Kelsey, Bruce Schneier, David Wagner and Chris Hall
- FSE ’98 Proceedings of the 5th International Workshop on Fast Software Encryption
- Pages 168-188

Cryptographic voting protocols: a systems perspective (paper)

- Chris Karlof, Naveen Sastry and David Wagner
- SSYM'05 Proceedings of the 14th conference on USENIX Security Symposium - Volume 14
- 17 Pages.
- https://www.usenix.org/legacy/event/sec05/tech/full_papers/karlof/karlof.pdf

Cyber Threats and Cyber Security (talk)

https://www.youtube.com/watch?v=agHkanpEfik  1:04:48

Designing security into software (MIT thesis)

- Zhang, Chang Tony
- https://dspace.mit.edu/handle/1721.1/35098


- Nathaniel Husted and Steven Myers
- Pages 1-14
- https://dl.acm.org/citation.cfm?doid=2683467.2683468

Formal methods for web security (paper -- survey)

- Michele Bugliesi, Stefano Calzavara, and Riccardo Focardi
- Journal of Logical and Algebraic Methods in Programming
- Volume 87, February 2017, Pages 110-126


- How Do System Administrators Resolve Access-Denied Issues in the Real World?
- Tianyin Xu, Han Min Naing, Le Lu and Yuanyuan Zhou
- http://dl.acm.org/citation.cfm?doid=3025453.3025999

Information Security at Western (Web Site)

http://security.uwo.ca/

Maroochy water breach (talk)

https://www.youtube.com/watch?v=C_PRhTXp6VQ  7:46

Measuring the security impacts of password policies using cognitive behavioral agent-based modeling (paper)

- Vijay Kothari, Jim Blythe, Sean W. Smith and Ross Koppel
- 9 pages.
- https://dl.acm.org/citation.cfm?doid=2746194.2746207

General Requirements of a Hybrid-Modeling Framework for Cyber Security (paper)

- Alessandro Oltramari, Noam Ben-Asher, Lorrie Cranor, Lujo Bauer and Nicolas Christin
- MILCOM ’14 Proceedings of the 2014 IEEE Military Communications Conference
- Pages 129-135

Model Checking An Entire Linux Distribution for Security Violations (paper)

- Benjamin Schwarz, Hao Chen, David Wagner, Jeremy Lin, Wei Tu, Geoff Morrison and Jacob West
- ACSAC ’05 Proceedings of the 21st Annual Computer Security Applications Conference
New Cyber Frontiers (video)

https://www.youtube.com/watch?v=eeE0_WZ55c0  37:41
The Agenda with Steve Paikin, Mar 1, 2018
(cybersecurity issues in recent federal budget)

One bit flips, one cloud flops: Cross-vm row hammer attacks and privilege escalation (paper)

- Xiao, Yuan and Zhang, Xiaokuan and Zhang, Yinqian and Teodorescu, Radu
- 25th USENIX Security Symposium, 2016, 17 pages

One Technique is Not Enough: A Comparison of Vulnerability Discovery Techniques (paper)

- Andrew Austin ; Laurie Williams
- Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on
- pp 97 – 106.

Privacy Impacts of IoT Devices: A SmartTV Case Study (paper)

- Richard L. Rutledge ; Aaron K. Massey ; Annie I. Antón
- Requirements Engineering Conference Workshops (REW), IEEE International
- 12-16 Sept. 2016
- pp. 261 -- 270

Risk Assessment for IoT : a system evaluation of the smart home and its cybersecurity imperative (MIT thesis)

- Gao, Olivia Qing
- https://dspace.mit.edu/handle/1721.1/106247

Schneier on Security (blog)

https://www.schneier.com/

Security collapse in the HTTPS market (paper)

Security collapse in the HTTPS market
Axel Arnbak, Hadi Asghari, Michel Van Eeten and Nico Van Eijk
Communications of the ACM, Volume 57 Issue 10, October 2014, Pages 47-55
http://dl.acm.org/citation.cfm?doid=2661061.2660574
see also:

Security Economics in the HTTPS Value Chain
Hadi Asghari, Michel van Eeten, Axel Arnbak, and N.A.N.M. van Eijk

and

Certificate Authority Collapse: Regulating Systemic Vulnerabilities in the HTTPS Value Chain
Axel Arnbak and N.A.N.M. van Eijk
2012 TRPC, 31 Pages

How Internet Resources Might Be Helping You Develop Faster but Less Securely (paper)

- Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L. Mazurek and Christian Stransky
- IEEE Security and Privacy
- Volume 15 Issue 2, April 2017
- Page 50-60

The most dangerous code in the world: validating SSL certificates in non-browser software (paper)

- Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov
- CCS ’12 Proceedings of the 2012 ACM conference on Computer and communications security
- Pages 38-49
- http://dl.acm.org/citation.cfm?doid=2382196.2382204

The web won’t be safe or secure until we break it (paper)

- Jeremiah Grossman
- Communications of the ACM
- Volume 56 Issue 1, January 2013
- Pages 68-72
- http://dl.acm.org/citation.cfm?doid=2398356.2398373

Security engineering: a guide to building dependable distributed systems (book)

- Security engineering: a guide to building dependable distributed systems
- by Anderson, Ross
- 2008, 2nd ed.
- QA76.9.A25A54 2008, Taylor Library Stack 6 (S6) - Regular Loan
- http://alpha.lib.uwo.ca/record=b4509089

free online copy at: http://www.cl.cam.ac.uk/~rja14/book.html

Selecting Security Patterns that Fulfill Security Requirements (paper)

- Michael Weiss and Haralambos Mouratidis
- RE ’08 Proceedings of the 2008 16th IEEE International Requirements Engineering Conference
- Pages 169-172

Software and the mundane management of air travel (paper)
Software and the mundane management of air travel
by Rob Kitchin and Martin Dodge
First Monday, special issue number 7 (September 2006),

Software-defined Networking enabled Resource Management and Security Provisioning in 5G Heterogeneous Networks (UWO Thesis)

- Software-defined Networking enabled Resource Management and Security Provisioning in 5G Heterogeneous Networks
- Xiaoyu Duan, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/4666/

An Access Control Model for NoSQL Databases (UWO Thesis)

- An Access Control Model for NoSQL Databases
- Motahera Shermin, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1797/

Secure OFDM System Design for Wireless Communications Hao Li, The University of Western Ontario (UWO Thesis)

- Secure OFDM System Design for Wireless Communications
- Hao Li, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/1343/

Security Protocol Suite for Preventing Cloud-based Denial-of-Service Attacks (UWO Thesis)

- Security Protocol Suite for Preventing Cloud-based Denial-of-Service Attacks
- Marwan M. Darwish, The University of Western Ontario
- http://ir.lib.uwo.ca/etd/3392/

Software security testing (paper)

- Software security testing
- B. Potter ; G. McGraw

An overview of vulnerability assessment and penetration testing techniques (paper)

- An overview of vulnerability assessment and penetration testing techniques
- Sugandh Shah and B. M. Mehtre
  https://link.springer.com/article/10.1007/s11416-014-0231-x

Comprehensive study of web application attacks and classification (paper)

- Comprehensive study of web application attacks and classification
- Manju Khari ; Sonam ; Vaishali ; Manoj Kumar
- Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on, 6 pages.

Mitigating program security vulnerabilities: Approaches and challenges (paper)
Mitigating program security vulnerabilities: Approaches and challenges
Hossain Shahriar and Mohammad Zulkernine
http://dl.acm.org/citation.cfm?doid=2187671.2187673


- Pete Herzog
- 39 pages.
  http://cdn.preterhuman.net/texts/other/osstmm.pdf

Security Testing Fundamentals - Ken van Wyk (talk)

https://www.youtube.com/watch?v=PYwqyVII8IQ 1:32:44

Software penetration testing (paper)

- Software penetration testing
- B. Arkin ; S. Stender ; G. McGraw
- IEEE Security & Privacy ( Volume: 3, Issue: 1, Jan.-Feb. 2005 ), Page(s): 84 - 87

Software Vulnerability Discovery Techniques: A Survey (paper)

- Software Vulnerability Discovery Techniques: A Survey
- Bingchang Liu ; Liang Shi ; Zuhua Cai ; Min Li

VCCFinder: Finding Potential Vulnerabilities in Open-Source Projects to Assist Code Audits (paper)

- Henning Perl, Sergej Dechand, Matthew Smith, Daniel Arp, Fabian Yamaguchi, Konrad Rieck, Sascha Fahl and Yasemin Acar
- CCS '15 Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security
  Pages 426-437
  https://dl.acm.org/citation.cfm?id=2813604


The Red Book: A Roadmap for Systems Security Research
Evangelos Markatos ; Davide Balzarotti ; Magnus Almgren ; Elias Athanasopoulos ; Herbert Bos ; Lorenzo Cavallaro ; Sotiris Ioannidis ; Martina Lindorfer ; Federico Maggi ; Zlatogor Minchev ; Farnaz Moradi ; Christian Platzer ; Iason Polakis ; Michalis Polychronakis ; Asia Slowinska ; Philippas Tsigas ; Stefano Zanero
195 pages.
http://publications.lib.chalmers.se/records/fulltext/188847/local_188847.pdf

Why security testing is hard (paper)

- Why security testing is hard
- H.H. Thompson
Survey on JavaScript security policies and their enforcement mechanisms in a web browser (paper)

- Nataliia Bielova
- The Journal of Logic and Algebraic Programming
- Volume 82, Issue 8, November 2013, Pages 243-262

Surviving the Web: A Journey into Web Session Security (paper)

- Stefano Calzavara, Riccardo Focardi, Marco Squarcina and Mauro Tempesta
- ACM Computing Surveys (CSUR)
- Volume 50 Issue 1, April 2017, 34 pages.
- https://dl.acm.org/citation.cfm?doid=3058791.3038923

The Internet of Things: Implications for Consumer Privacy under Canadian Law (paper)

- Samuel E. Trosow, Lindsay Taylor and Alexandrina Hanam,
- 2017
- http://ir.lib.uwo.ca/lawpub/91/

Timing Analysis of Keystrokes and Timing Attacks on SSH (paper)

Timing Analysis of Keystrokes and Timing Attacks on SSH
Dawn Xiaodong Song, David Wagner and Xuqing Tian
USENIX Security’01, 16 pages
https://www.usenix.org/legacy/events/sec2001/song.html

Practical Keystroke Timing Attacks in Sandboxed JavaScript. (paper)

- Lipp M., Gruss D., Schwarz M., Bidner D., Maurice C., Mangard S.
- page 191 - 209.
- https://link.springer.com/chapter/10.1007/978-3-319-66399-9_11

Using a systems-theoretic approach to analyze cyber attacks on cyber-physical systems (MIT thesis)

Whyte, David L., 1967-
Thesis: S.M. in Engineering and Management, Massachusetts Institute of Technology, School of Engineering, System Design and Management Program, 2017
https://dspace.mit.edu/handle/1721.1/110143

Using STAMP to understand recent increases in malicious software activity (MIT thesis)

Zipkin, David S
https://dspace.mit.edu/handle/1721.1/32285

Verifying computations without reexecuting them (paper)

- Michael Walfish and Andrew J. Blumberg
- Communications of the ACM
- Volume 58 Issue 2, February 2015
- Pages 74-84
What Every Engineer Needs to Know About Security and Where to Learn It (talk)

https://www.youtube.com/watch?v=LL_n7dIBgY 49:05
see: https://www.privacyrights.org/data-breaches for one list of known data breaches of corporate/government web sites

Why cryptosystems fail (paper)

- Ross J. Anderson
- Communications of the ACM
- Volume 37 Issue 11, Nov. 1994
- Pages 32-40
- https://dl.acm.org/citation.cfm?doid=188280.188291

EMV: why payment systems fail (paper)

- Ross Anderson and Steven J. Murdoch
- Communications of the ACM CACM Homepage archive
- Volume 57 Issue 6, June 2014
- Pages 24-28
- https://dl.acm.org/citation.cfm?doid=2602695.2602321

Why information security is hard - an economic perspective (paper)

- R. Anderson
- 8 pages

Technology Acceptance as a non-functional requirement

A frequent issue with software is whether or not the users want to use it once it has been built. While such a requirement is often implicit, it is worthwhile giving some thought to what would be necessary to design software that people didn't avoid whenever possible.

- Technology acceptance model: a literature review from 1986 to 2013 (paper)
- A computational literature review of the technology acceptance model (paper)
- The Technology Integration Model (TIM). Predicting the continued use of technology (paper)

There are connections between acceptability and usability of a system (cf Usability as a non-functional requirement)

- Technology Acceptance and User Experience: A Review of the Experiential Component in HCI (paper)

A computational literature review of the technology acceptance model (paper)

- Michael J. Mortenson and Richard Vidgen
- International Journal of Information Management
- Volume 36, Issue 6, Part B, December 2016, Pages 1248-1259
- https://doi.org/10.1016/j.ijinfomgt.2016.07.007

Technology Acceptance and User Experience: A Review of the Experiential Component in HCI (paper)
Technology acceptance model: a literature review from 1986 to 2013 (paper)

Maranguni, N. & Grani, A.
Universal Access in the Information Society
March 2015, Volume 14, Issue 1, pp 81–95
https://doi.org/10.1007/s10209-014-0348-1

The Technology Integration Model (TIM). Predicting the continued use of technology (paper)

Heather Shaw, David A. Ellis, Fenja V.Ziegler
Computers in Human Behavior
Volume 83, June 2018, Pages 204-214
https://doi.org/10.1016/j.chb.2018.02.001

Usability as a non-functional requirement

General overview:
- Architecting for usability: a survey (paper)
- Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience (paper)

A business view of justifying usability:
- A Strategic Approach to Metrics for User Experience Designers (paper)

More general development of usability metrics:
- Measuring the user experience on a large scale: user-centered metrics for web applications (paper)
- Predictive human performance modeling made easy (paper)
- Deploying CogTool: integrating quantitative usability assessment into real-world software development (paper)
- SUS: a retrospective (paper)
- The Usability Metric for User Experience (paper)

One approach to usability is user surveys
- IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use (paper)

Usability studies:
- Why Johnny can’t encrypt: A usability evaluation of PGP 5.0 (paper)
- Towards Enhanced Usability of IT Security Mechanisms - How to Design Usable IT Security Mechanisms Using the Example of Email Encryption (paper)
- Reverse engineering of content to find usability problems: a healthcare case study (paper)

Journals and Conferences on Usability
- Journal of Usability Studies (journal)

Architecting for usability: a survey (paper)

Eelke Folmer and Jan Bosch
Journal of Systems and Software
Volume 70, Issues 1–2, February 2004, Pages 61-78
A Strategic Approach to Metrics for User Experience Designers (paper)

- Carl W. Turner
- Journal of Usability Studies
- Volume 6 Issue 2, February 2011
- Pages 52-59
- https://dl.acm.org/citation.cfm?id=2010382

IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use (paper)

- James R. Lewis
- http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.584.6610

Journal of Usability Studies (journal)

- Usability Professionals’ Association
- https://dl.acm.org/citation.cfm?id=3173069

Measuring the user experience on a large scale: user-centered metrics for web applications (paper)

- Kerry Rodden, Hilary Hutchinson and Xin Fu
- CHI ’10 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems
- Pages 2395-2398
- https://dl.acm.org/citation.cfm?doid=1753326.1753687

Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience (paper)

- Javier A. Bargas-Avila and Kasper Hornbæk
- Pages 2689-2698
- https://dl.acm.org/citation.cfm?doid=1978942.1979336

Predictive human performance modeling made easy (paper)

- Bonnie E. John, Konstantine Prevas, Dario D. Salvucci and Ken Koedinger
- CHI ’04 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems
- Pages 455-462
- https://dl.acm.org/citation.cfm?doid=985692.985750

Deploying CogTool: integrating quantitative usability assessment into real-world software development (paper)

- Rachel Bellamy, Bonnie John and Sandra Kogan
- ICSE ’11 Proceedings of the 33rd International Conference on Software Engineering
- Pages 691-700
- https://dl.acm.org/citation.cfm?doid=1985793.1985890
Reverse engineering of content to find usability problems: a healthcare case study (paper)

- Shadi Ghajar-Khosravi, Flora Wan, Samir Gupta and Mark Chignell
- Journal of Usability Studies, Volume 8 Issue 1, November 2012, Pages 16-28
- https://dl.acm.org/citation.cfm?id=2817911

SUS: a retrospective (paper)

- John Brooke
- Journal of Usability Studies, Volume 8 Issue 2, February 2013, Pages 29-40
- https://dl.acm.org/citation.cfm?id=2817913

The Usability Metric for User Experience (paper)

- Kraig Finstad
- Interacting with Computers, Volume 22, Issue 5, 1 September 2010, Pages 323-327
- https://doi.org/10.1016/j.intcom.2010.04.004

Why Johnny can’t encrypt: A usability evaluation of PGP 5.0 (paper)

- A. Whitten and J. D. Tygar.
- (earlier version of this paper was published in Proceedings of the 8th USENIX Security Symposium, August 1999, pp. 169-183.
  https://people.eecs.berkeley.edu/~tygar/papers/Why_Johnny_Cant_Encrypt/USENIX.pdf)

Towards Enhanced Usability of IT Security Mechanisms - How to Design Usable IT Security Mechanisms Using the Example of Email Encryption (paper)

- Hans-Joachim Hof
- International Journal On Advances in Security, volume 6, number 1&2, pp. 78-87 ISSN 1942-2636, 2013

Writing papers in computer science

Whether you are working on a programming task, analyzing an algorithm, or doing a literature investigation, ultimately, people are going to want to know what you learned from it and thought about it – and this involves some sort of write up or presentation. As noted by Lesley Lamport (recent Turing Award winner) in his talk Leslie Lamport: Thinking Above the Code https://www.youtube.com/watch?v=-4Yp3j_jk8Q at 3:27, "To think you have to write. If you're thinking without writing, you only think you're thinking."

Mostly I am interested in writing, but some notes regarding presentations are:

- YOW! Nights October 2014 Damian Conway - Instantly Better Presentations (talk)
- Evaluating oral presentations (paper)

Writing anything significant is a project and all we know about project management is relevant to the task. In particular:

- Don't Go Dark (blog entry)

An issue with regards to research that is getting increasing attention is whether or not the results are reproducible

- Reproducibility of Results in the ACM Digital Library (website)
- Editorial: ACM TOMS Replicated Computational Results Initiative (paper)
- Proceedings of the Reproducibility Workshop 2017 (conference)
- Challenges with Reproducibility (paper)

Thanks to the university library systems digital subscription to Springer Verlag, some useful books for writing related to computer science are available online:

- Writing for Computer Science [electronic resource] / by Justin Zobel (book) – note chapter 3 on Reading and Reviewing
Some notes on how to get started:

- Framing a research study (paper)
- Research methods in computing: what are they, and how should we teach them? (paper)

There are different kinds of writings:

- Writing for synthesis of evidence in empirical software engineering (paper)
- Software engineering article types: An analysis of the literature (paper)
- Presenting software engineering results using structured abstracts: a randomised experiment (paper)
- Case studies synthesis: a thematic, cross-case, and narrative synthesis worked example (paper)
- What Does It Mean to “Measure Performance”? (paper)

One type of writing task that fits the time frame of a course is a literature review

- Using qualitative metasummary to synthesize empirical findings in literature reviews (paper)
- How to Write a Literature Review Paper? (paper)
- The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature (paper)
- Performing systematic literature reviews in software engineering (paper)
- A critical appraisal of systematic reviews in software engineering from the perspective of the research questions asked in the reviews (paper)
- Experience-based guidelines for effective and efficient data extraction in systematic reviews in software engineering (paper)
- Outcomes of a community workshop to identify and rank barriers to the systematic literature review process (paper)
- Quality assessment of systematic reviews in software engineering: a tertiary study (paper)
- Strength of evidence in systematic reviews in software engineering (paper)
- Systematic literature reviews and undergraduate research (paper)
- Systematic review in software engineering: where we are and where we should be going (paper)
- Writing for synthesis of evidence in empirical software engineering (paper)

The literature is often large, leading to the question of how do you find what you need?

- A critical analysis of studies that address the use of text mining for citation screening in systematic reviews (paper)
- Guidelines for snowballing in systematic literature studies and a replication in software engineering (paper)
- Experiences from using snowballing and database searches in systematic literature studies (paper)
- Second-generation systematic literature studies using snowballing (paper)
- Systematic literature studies: database searches vs. backward snowballing (paper)
- Using Forward Snowballing to update Systematic Reviews in Software Engineering (paper)

A variation on the literature review is the systematic mapping

- Systematic mapping studies in software engineering (paper)
- Guidelines for conducting systematic mapping studies in software engineering: An update (paper)
- Improving Systematic Mapping Reviews (paper)
- Using mapping studies as the basis for further research - A participant-observer case study (paper)

And there is always the question of how to do the bibliography

- How Bibliographies, Citations, etc. are Handled in Computer Science (note)

Another issue that arises is what sort of evidence is needed to justify a claim

- Greg Wilson - What We Actually Know About Software Development, and Why We Believe It’s True (talk)
- KOLL-2016-11-25 - Views on external and internal validity in software engineering (Janet Siegmund) (talk)
- NYT E03 (2016) - Controlled Experiments (Siegmund) (talk)

Case studies synthesis: a thematic, cross-case, and narrative synthesis worked example (paper)

- Case studies synthesis: a thematic, cross-case, and narrative synthesis worked example
- Daniela S. Cruzes, Tore Dybå, Per Runeson and Martin Höst
- https://link.springer.com/article/10.1007%2Fs10664-014-9326-8

Guidelines for snowballing in systematic literature studies and a replication in software engineering (paper)

- Guidelines for snowballing in systematic literature studies and a replication in software engineering
Experiences from using snowballing and database searches in systematic literature studies (paper)

Second-generation systematic literature studies using snowballing (paper)

Systematic literature studies: database searches vs. backward snowballing (paper)

Using Forward Snowballing to update Systematic Reviews in Software Engineering (paper)

Performing systematic literature reviews in software engineering (paper)

A critical analysis of studies that address the use of text mining for citation screening in systematic reviews (paper)

A critical appraisal of systematic reviews in software engineering from the perspective of the research questions asked in the reviews (paper)
- A critical appraisal of systematic reviews in software engineering from the perspective of the research questions asked in the reviews
- Fabio Q. B. da Silva, André L. M. Santos, Sérgio C. B. Soares, A. César C. França and Cleviton V. F. Monteiro
  - http://dl.acm.org/citation.cfm?doid=1852786.1852830

Experience-based guidelines for effective and efficient data extraction in systematic reviews in software engineering (paper)

- Experience-based guidelines for effective and efficient data extraction in systematic reviews in software engineering
- Vahid Garousi and Michael Felderer
  - http://dl.acm.org/citation.cfm?doid=3084226.3084238

Outcomes of a community workshop to identify and rank barriers to the systematic literature review process (paper)

- Outcomes of a community workshop to identify and rank barriers to the systematic literature review process
- Edgar Hassler, Jeffrey C. Carver, Nicholas A. Kraft and David Hale
- EASE '14 Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, Article No. 31
  - http://dl.acm.org/citation.cfm?doid=2601248.2601274

Quality assessment of systematic reviews in software engineering: a tertiary study (paper)

- Quality assessment of systematic reviews in software engineering: a tertiary study
- You Zhou, He Zhang, Xin Huang, Song Yang, Muhammad Ali Babar and Hao Tang
  - http://dl.acm.org/citation.cfm?doid=2745802.2745815

Strength of evidence in systematic reviews in software engineering (paper)

- Strength of evidence in systematic reviews in software engineering
- Tore Dybå and Torgeir Dingsøyr
- ESEM '08 Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement, Pages 178-187
  - http://dl.acm.org/citation.cfm?doid=1414004.1414034

Systematic literature reviews and undergraduate research (paper)

- Systematic literature reviews and undergraduate research
- Tony Clear
- ACM Inroads, Volume 3 Issue 4, December 2012, Pages 10-11
  - http://dl.acm.org/citation.cfm?doid=2381083.2381087

Systematic review in software engineering: where we are and where we should be going (paper)

- Systematic review in software engineering: where we are and where we should be going
- Barbara A. Kitchenham
  - http://dl.acm.org/citation.cfm?doid=2372233.2372235
Presenting software engineering results using structured abstracts: a randomised experiment (paper)

- Presenting software engineering results using structured abstracts: a randomised experiment
- David Budgen, Barbara A. Kitchenham, Stuart M. Charters, Mark Turner, Pearl Brereton and Stephen G. Linkman

Software engineering article types: An analysis of the literature (paper)

- Software engineering article types: An analysis of the literature
- Michela Montesi and Patricia Lago

Systematic mapping studies in software engineering (paper)

- Systematic mapping studies in software engineering
- Kai Petersen, Robert Feldt, Shahid Mujtaba and Michael Mattsson
- EASE'08 Proceedings of the 12th international conference on Evaluation and Assessment in Software Engineering, Pages 68-77

Guidelines for conducting systematic mapping studies in software engineering: An update (paper)

- Guidelines for conducting systematic mapping studies in software engineering: An update
- Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz
- Information and Software Technology, Volume 64, August 2015, Pages 1-18

Improving Systematic Mapping Reviews (paper)

- Improving Systematic Mapping Reviews
- Alberto Sampaio
- ACM SIGSOFT Software Engineering Notes, Volume 40 Issue 6, November 2015, Pages 1-8
- http://dl.acm.org/citation.cfm?doid=2830719.2830732

Using mapping studies as the basis for further research - A participant-observer case study (paper)

Using mapping studies as the basis for further research - A participant-observer case study
Barbara A. Kitchenham, David Budgen and O. Pearl Brereton
Information and Software Technology, Volume 53 Issue 6, June, 2011, Pages 638-651

The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature (paper)

- The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature
- Vahid Garousi, Michael Felderer and Mika V. Mäntylä
- EASE '16 Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering, Article No. 26
- http://dl.acm.org/citation.cfm?doid=2915970.2916008
Using qualitative metasummary to synthesize empirical findings in literature reviews (paper)

- Using qualitative metasummary to synthesize empirical findings in literature reviews
- Danilo Monteiro, Marcos Cardoso, Fabio Q. B. da Silva and César França
- http://dl.acm.org/citation.cfm?doid=2652524.2652562

Writing for synthesis of evidence in empirical software engineering (paper)

- Writing for synthesis of evidence in empirical software engineering
- Claes Wohlin
- ESEM '14 Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, Article No. 46
- http://dl.acm.org/citation.cfm?doid=2652524.2652559

Framing a research study (paper)

- Framing a research study
- Tony Clear
- ACM Inroads, Volume 5 Issue 2, June 2014, Pages 22-23
- http://dl.acm.org/citation.cfm?doid=2614512.2614517

Research methods in computing: what are they, and how should we teach them? (paper)

- Research methods in computing: what are they, and how should we teach them?
- Hilary J. Holz, Anne Applin, Bruria Haberman, Donald Joyce, Helen Purchase and Catherine Reed
- ITiCSE-WGR ’06 Working group reports on ITiCSE on Innovation and technology in computer science education, Pages 96-114
- http://dl.acm.org/citation.cfm?doid=1189136.1189180

YOW! Nights October 2014 Damian Conway - Instantly Better Presentations (talk)

https://www.youtube.com/watch?v=W_l_DrWic88  1:31:26

speaker's notes: http://damian.conway.org/IBP.pdf

Evaluating oral presentations (paper)

- Evaluating oral presentations
- Dennis S. Martin
- http://dl.acm.org/citation.cfm?id=1040203

What Does It Mean to “Measure Performance”? (paper)

- What Does It Mean to “Measure Performance”?
- Alistair Moffat and Justin Zobel
- https://link.springer.com/chapter/10.1007/978-3-540-30480-7_1
Writing for Computer Science [electronic resource] / by Justin Zobel (book)

- 3rd ed. 2014.
- Zobel, Justin
- Springer London
- http://alpha.lib.uwo.ca/record=b6478775

How to Write a Better Thesis [electronic resource] (book)

- Publisher Cham : Springer International Publishing : Imprint: Springer, 2014.
- http://alpha.lib.uwo.ca/record=b6314865~S20


- Thesis Projects: A Guide for Students in Computer Science and Information Systems
- Mikael Berndtsson, Jörgen Hansson, Björn Olsson, Björn Lundell
- Springer Verlag

Don't Go Dark (blog entry)

https://blog.codinghorror.com/dont-go-dark/

Although this blog entry specifically addresses issues in writing programs, the same issues arise in any substantial piece of work done in an academic context, for example, writing a paper. As long as your work is being constantly reviewed, you don’t have to worry about all the different ways you could go astray or about putting a lot of effort into something only to find it wasn’t what was wanted.

Reproducibility of Results in the ACM Digital Library (website)

http://dl.acm.org/docs/reproducibility.cfm

Challenges with Reproducibility (paper)

- Challenges with Reproducibility
- Vaibhav Bajpai, Mirja Kühlewind, Jörg Ott, Jürgen Schönwälder, Anna Sperotto and Brian Trammell
- Reproducibility ’17 Proceedings of the Reproducibility Workshop, Pages 1-4
- http://dl.acm.org/citation.cfm?doid=3097766.3097767

Editorial: ACM TOMS Replicated Computational Results Initiative (paper)

Editorial: ACM TOMS Replicated Computational Results Initiative
Michael A. Heroux
ACM Transactions on Mathematical Software (TOMS) TOMS Homepage archive
Volume 41 Issue 3, June 2015, 5 pages
http://dl.acm.org/citation.cfm?doid=2786970.2743015

Proceedings of the Reproducibility Workshop 2017 (conference)
How to Write a Literature Review Paper? (paper)

- How to Write a Literature Review Paper?
- Bert Van Wee and David Banister

How Bibliographies, Citations, etc. are Handled in Computer Science (note)

There is no one way to do these things in Computer Science. Generally different publishers, but sometimes even different journals, specify how they want things written. Three popular styles are

- ACM https://www.acm.org/publications/authors/submissions

In reading the literature in Computer Science, you will encounter many different styles. Within a given writing, only one style should be followed and that style should be followed consistently.

Greg Wilson - What We Actually Know About Software Development, and Why We Believe It’s True (talk)

https://vimeo.com/9270320 1:06:37

- 5:00 the way people program hasn't changed in the last 25 years
- 9:45 "if we ask a question carefully, and if we are willing to be humble enough to admit when we are wrong, if we look at the date, then we can find out how the universe actually works, and then we can do things based on that knowledge."
- references Martin Fowler's blog on Software Design https://www.martinfowler.com/
- 11:40 looks at quote "[Using domain-specific languages] leads to two primary benefits. The first, and simplest, is improved programmer productivity ... The second ... is ... communication with domain experts." A Pedagogical Framework for Domain-Specific Languages, M. Fowler, IEEE Software ( Volume: 26, Issue: 4, July-Aug. 2009 ), http://ieeexplore.ieee.org/document/5076452/
- two substantial claims of fact in an academic journal without a single citation/evidence
- low standards of proof
- references work of Lutz Prechelt on empirically evaluating productivity variations between programmers, effects of language, effects of web programming frameworks claims https://scholar.google.ca/citations?user=ZY2MP7QAAAAJ&hl=en&oi=ao&cstart=0&pagesize=20
- sidenote:
- sidenotis
- results:
  1) rather than 28:1, 14:1 is more accurate in terms of best programmer productivity versus worst
  2) rather than compare best to worst, better to compare median of worst quarter to median of best quarter, which varies significantly depending on task type, but is rarely larger than 4.
  22:48 output in terms of lines of code per hour is the same across languages
- 30:03 some major known results
- 1) for every 25% increase in problem complexity, there is a 100% increase in solution complexity (Woodfield, 1979) http://ieeexplore.ieee.org/document/1702600/ An Experiment on Unit Increase in Problem Complexity, S. N. Woodfield, IEEE Transactions on Software Engineering ( Volume: SE-5, Issue: 2, March 1979 ), Page(s): 76 - 79.
  3) If more than 20-25% of a component has to be revised, it's better to rewrite it from scratch (Thomas et al, 1997) Boeing avionics software

43:55 discusses Beautiful Code book series

Beautiful Code: Leading Programmers Explain How They Think – Jul 6 2007; by Oram (Editor), Wilson (Editor); https://www.amazon.ca

note: another overview of this talk can be found in http://www.gdb.me/computing/citations-greg-wilson-cusec.html

note: another overview of this talk can be found at https://softwareflow.wordpress.com/2011/11/10/greg-wilsons-what-we-actually-know-about-software-development/

further discussion on this topic appears in:

https://www.youtube.com/watch?v=qDY-Lk1rsQY 1:52:03 RR 184 - What We Actually Know About Software Development and Why We Believe It's True - Ruby Rogues

• a bunch of ads up til 2:46
• then a bunch of discussion on the topic (extensive notes on the youtube page)
• 11:30 neat quote: most people would rather fail than change

KOLL-2016-11-25 - Views on external and internal validity in software engineering (Janet Siegmund) (talk)

https://www.youtube.com/watch?v=ueYQoUZiA7g 43:06

• 0:36 consider how you would try to answer the research question: What is more productive to use, a statically or dynamically typed language?
• internal validity: tightly controlled experiment that doesn't look much like real world
• external validity: very realistic data but can't tell what is actually causing the effect (confounding parameters: many different things that could impact your results)
• looked at 450 papers to see how authors addressed this tradeoff
• 94% of the papers analyzed used empirical methods
• 8% reported on a replication study (repeat experiment with controlled variation)
• 46% of papers did not mention explicitly 'threats to validity' (parameters that weren't controlled for but might have impacted result)
• note: reader doesn't usually know enough about details of experiment to guess threats on their own
• 10:41 The NIPS experiment
• program committees review submitted papers and put best in conference.
• experiment split committee in half and had each half do a review of 166 submissions.
• results were as if the review committees were just randomly (weighted by a few basic properties that most papers had) selecting papers
• 12:00 survey of reviews re views on validity
• wide range of reviewer opinion on whether internal validity was pointless in software engineering while others thought it was critical.
• some reviewers think each study has to have immediate practical impact whereas others think you have to build up pieces of a puzzle to get a significant result
• most reviewers see need of replication results but think they are hard to get published and difficult to do
• some reviewers confound external validity with practicality -- external validity is about generalizability
• bottom line (quote from one reviewer who participated in survey): “We do not know what we are doing.”
• 25:48 so what to do about all this
• ‘educate’ reviewers
• develop standards for review
authors need to conduct multiple studies some internally valid and externally valid and report on validity issues and be concrete
28:19 all data related to this study at http://www.infosun.fim.uni-passau.de/spl/janet/ese/
28:37 questions
sidenote:
Actually, it also turns out to be an issue in medicine as well.
Why Most Published Research Findings Are False, John P. A. Ioannidis, August 30, 2005, http://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.0020124
more generally
http://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970
etonedis:
NYT E03 (2016) - Controlled Experiments (Siegmund) (talk)

https://www.youtube.com/watch?v=HyYxq2UV4Ug 1:25:59

2:00 what are experiments?
* systematic research study
* one or more factors intentionally varied
* everything else held constant
* result of systematic variation is observed
the focus of this talk is on experiments involving humans
such experiments are often done in psychology
Wilhelm Wundt (1832 - 1920) is regarded as "the father of experimental psychology" https://en.wikipedia.org/wiki/Wilhelm_Wundt
Stages of Experiments:
* objective definition (hypothesis; independent and dependent variables)
* design (experimental design, confounding variables)
* execution (data)
* analysis (accepted/rejected hypothesis)
* interpretation
Running example re comments in source code
* do they make code more comprehensible?
* independent variable, predictor that you will intentionally vary and influences the dependent variables
* here the independent variable is the comments
things that could be measured: whether or not they are there, whether they are short or long, where they are located, are they good or bad comments
dependent variable would be comprehensibility of the code
how do you measure comprehensibility? development time is one approach, give quiz on whether people understood code, ask participants to do subjective rating of comprehensibility, think aloud protocol (https://en.wikipedia.org/wiki/Think_aloud_protocol)
sidenote: the think aloud protocol is based on Protocol analysis https://en.wikipedia.org/wiki/Protocol_analysis which was originally developed (On the analysis of human problem solving protocols, Allen Newell, 1966 http://repository.cmu.edu/cgi/viewcontent.cgi?article=2941&context=compsci ) by AI researchers who were trying to understand how humans do problem solving to get ideas about how computers should do problem solving
etonedis
10:44 hypotheses
expectations about the outcome and why you have these expectations
hypothesis must be falsifiable (Karl Popper's The Logic of Scientific Discovery, 1934/1959 https://en.wikipedia.org/wiki/The_Logic_of_Scientific_Discovery) example hypothesis: comments describing each statement of source code have no effect on response time of understanding source code
comments describing the purpose of statements speed up comprehension
hypothesis are needed because they influence how the experiment is designed
avoid "fishing for results" https://en.wikipedia.org/wiki/Testing_hypotheses_suggested_by_the_data
when doing exploring, need to do further studies to verify what you think you found (your new hypothesis)
25:30 validity (do we measure what we want to measure?)
26:40 internal: degree to which the value of the dependent variable can be assigned to the manipulation of the independent variable
27:30 external: degree to which the results gained in one experiment can be generalized to other participants and settings
control for confounding variables https://en.wikipedia.org/wiki/Confounding by randomization, matching, holding constant, making it an independent variable, and analyzing its influence on result
38:00 which control technique for confounding variables is best?
basically this is all done in multiple studies. first maximize internal validity and then start looking at confounding variables and how results can be generalizable
49:46 experimental design
• how you organize participants into groups and how to organize tasks the groups do
• 55:34 execution
• what can go wrong? do pilot tests https://en.wikipedia.org/wiki/Pilot_experiment
• tell participants exactly what they have to do
• observe that participants do what they are instructed to do
• make backups of data
• 59:39 ethics
• 1:02:00 analysis of data
• 1:13:15 interpretation of results
• need to wonder how time to write the comments relates to speedup in comprehension time
• 1:17:44 and now?
• every experiment has flaws -- so look for what is best to do in your case
• 1:21:22 questions