Software Verification and Validation (CSE 565)

Note: The information below and course outline are subject to modifications and updates.

About this Course
Software as a stand alone product or embedded within a system plays an integral role in our world today. As a consequence, it is essential that software works as expected. This requires software testing which entails answering both the verification question: “Are we building the product right?” and the validation question: “Are we building the right product?”. Understanding these questions is crucial for developing good test cases. This course is for anyone involved in testing software at any level starting from code modules to system testing. Strategies and techniques are presented for both testing software as well as planning and tracking testing efforts.

Specific topics covered include:
- Testing background
- Testing process activities
- Requirements based testing techniques
- Structure based testing techniques
- System testing
- Testing tools
- Reliability models
- Statistical testing
- Test planning
- Tracking testing progress
- Test documentation
- Test process improvement

Required Prior Knowledge and Skills
- High-level programming language
- Software development life cycle models

Learning Outcomes
Learners completing this course will be able to:
- Explain how testing activities fit within leading software development process models
- Understand and apply best practices for software testing
- Create test cases based on commonly used requirements based testing techniques
- Create test cases to achieve control and data flow structure based coverage
- Apply static analysis techniques to identify code anomalies
- Create test cases that demonstrate system-level quality requirements are being met
- Identify appropriate testing tools for applications
Major Project

**Title:** Specification-Based Testing

**Description:** Use specification based testing strategies to develop and execute tests to identify defects in given code and record defects in the given defect tracking spreadsheet.

**Multi-part?** No  **Team-based?** No

**Resulting portfolio artifacts:** Set of test cases and defect tracking spreadsheet

**Real-world/employer relevance:** Practical application of testing techniques to real problem

**Title:** Specification-Based Testing: Design of Experiments

**Description:** Given a set of requirements, design a set of test cases following the design of experiments technique.

**Multi-part?** No  **Team-based?** No

**Resulting portfolio artifacts:** Set of test cases

**Real-world/employer relevance:** Practical application of testing techniques to real problem

**Title:** Structural Based Testing

**Description:** Use a structural testing tool to develop and execute tests on given code to achieve desired coverage level.

**Multi-part?** No  **Team-based?** No

**Resulting portfolio artifacts:** Set of test cases and tool coverage report

**Real-world/employer relevance:** Practical application of testing techniques to real problem

**Title:** Risk Based Testing

**Description:** Given a real life scenario, utilize risk based testing techniques to evaluate and prioritize tasks.

**Multi-part?** No  **Team-based?** No

**Resulting portfolio artifacts:** Tasklist prioritized by risk

**Real-world/employer relevance:** Practical application of analyzing and using risk based testing techniques to real life scenario

Course Content

**Instruction**

- Video lectures
- Other videos (animations, demos, etc.)
- Readings
- Live sessions (webinars, virtual office hours)

**Assessments**

- Quizzes (auto-graded)
- Unit assignments (peer-reviewed or auto-graded)
- Graded discussion
- Project (team and/or individual, instructor-graded)
- Exam(s) (proctored, auto- and/or instructor-graded)
Estimated Workload/ Time Commitment Per Week
15 - 20 hours per week

Technology Requirements
Software and Other (programs, platforms, services, etc.)
Most technology integrations will be provided through Coursera

Course Outline

Unit 1: Software Testing Overview
Module 1: Testing Background
Module 2: Testing Throughout Life Cycle
Module 3: Testing Best Practices and Standards

Unit 2: Specification-based Testing Technique Part 1
Module 1: Input Sampling Techniques
Module 2: Model-Based Testing

Unit 3: Specification-based Testing Technique Part 2
Module 1: Combinatorial Testing Techniques
Module 2: Mutation Testing
Module 3: Fuzz Testing
Module 4: Metamorphic Testing
Module 5: Defect Based Testing
Module 6: Exploratory Testing

Unit 4: Structural Based Testing Strategies
Module 1: Dynamic Analysis
Module 2: Static Analysis

Unit 5: Testing Software Quality Characteristics Part 1
Module 1: Performance Testing
Module 2: Stress Testing
Module 3: Volume Testing
Module 4: Configuration Testing
Module 5: Regression Testing
Module 6: Mobile Testing
Unit 6: Testing Software Quality Characteristics Part 1
Module 1: Error Detection, Recovery and Serviceability Testing
Module 2: Usability Testing
Module 3: Reliability Testing

Unit 7: Test Management Part 1
Module 1: Testing Plan
Module 2: Testing Schedule
Module 3: Estimating Testing Effort
Module 4: Risk Based Testing
Module 5: Test Exit Criteria
Module 6: Test Documentation

Unit 8: Test Management Part 2
Module 1: Test Tracking
Module 2: Test Process Improvement
Module 3: Test Outsourcing
Module 4: People Management
Module 5: Software Inspections
Module 6: Causal Analysis
Module 7: Test Maturity Model

About ASU
Established in Tempe in 1885, Arizona State University (ASU) has developed a new model for the American Research University, creating an institution that is committed to access, excellence and impact.

As the prototype for a New American University, ASU pursues research that contributes to the public good, and ASU assumes major responsibility for the economic, social and cultural vitality of the communities that surround it. Recognizing the university’s groundbreaking initiatives, partnerships, programs and research, U.S. News and World Report has named ASU as the most innovative university all three years it has had the category.

The innovation ranking is due at least in part to a more than 80 percent improvement in ASU’s graduation rate in the past 15 years, the fact that ASU is the fastest-growing research university in the country and the emphasis on inclusion and student success that has led to more than 50 percent of the school’s in-state freshman coming from minority backgrounds.

About Ira A. Fulton Schools of Engineering
Structured around grand challenges and improving the quality of life on a global scale, the Ira A. Fulton Schools of Engineering at Arizona State University integrates traditionally separate disciplines and supports collaborative research in the multidisciplinary areas of biological and health systems; sustainable engineering and the built environment; matter, transport and energy; and computing and decision systems. As the largest engineering program in the United States, students can pursue their educational and career goals through 25 undergraduate degrees or 39 graduate programs and rich experiential education offerings. The Fulton Schools are dedicated to engineering programs that combine a strong core foundation with top faculty and a reputation for graduating students who are aggressively recruited by top companies or become superior candidates for graduate studies in medicine, law, engineering and science.
About the School of Computing, Informatics, & Decision Systems Engineering

The School of Computing, Informatics, and Decision Systems Engineering advances developments and innovation in artificial intelligence, big data, cybersecurity and digital forensics, and software engineering. Our faculty are winning prestigious honors in professional societies, resulting in leadership of renowned research centers in homeland security operational efficiency, data engineering, and cybersecurity and digital forensics. The school’s rapid growth of student enrollment isn’t limited to the number of students at ASU’s Tempe and Polytechnic campuses as it continues to lead in online education. In addition to the Online Master of Computer Science, the school also offers an Online Bachelor of Science in Software Engineering, and the first four-year, completely online Bachelor of Science in Engineering program in engineering management.

Creator

Jim Collofello

Jim Collofello serves as Vice Dean of Academic and Student Affairs and has held this position since 2006. In this capacity he leads the school’s student recruitment and retention, career development and placement, K-12 programming, new curriculum development, accreditation and oversight of Fulton Difference programming. The Fulton Difference consists of innovative programs operated at scale to provide students with opportunities to develop and enhance their research, leadership, project development and entrepreneurship skills. Major Fulton Difference programs include engineering student organizations, Fulton Undergraduate Research Initiative, Grand Challenge Scholars Program, Undergraduate Teaching Assistant Program and Engineering Projects in Community Service. He is also a professor of computer science and software engineering. His teaching and research interests lie in the software engineering area with an emphasis on software quality assurance, software process improvement and software project management. He has also been active in developing and improving computer science curriculum and working to improve undergraduate retention. In addition to his academic activities, he has also been involved in applied research projects, training and consulting with many large corporations over the last 25 years.