CSE 464: Software Quality Assurance (SQA) and Testing (3)

School of Computing, Informatics, and Decision Systems Eng. Arizona State University

Catalog Description: Topics include software quality assurance, software quality matrices, software configuration management, software verification and validation, reviews, inspections, understanding software testing process, functional testing, structural testing, model-based testing, integration, system, and regression testing techniques, software life cycle models and software testing, testing distributed software, bug tracking, and use of testing tools.

Course Objectives and Expected Learning Outcomes: By the end of this course, students should be able to

- *explain* fundamental SQA considerations and understand the role of testing in SQA
- *explain* the basis of various approaches in unit, integration, and system testing
- *understand* how testing is applied throughout the software life cycle
- *able* to use an automated testing tool to write test cases and perform testing
- *carryout* test driven development for complex software using testing tools
- *able* to perform inspections, reviews, and walkthroughs of code and design of a large software system
- *explain* basic principles of software configuration management
- gain a solid understanding of state of the art in software testing

Requirement/Prerequisites: CSE 360 C or better (or equivalent)

Major Topics Covered in the Course:

- Basic concepts in software quality assurance
- Software testing as a quality assurance technique
- Principles of testing, testing process activities, testing standards, and matrices
- Functional testing
- Model-based testing
- Structure based testing
- Automated unit testing and test driven development
- Refactoring and Regression testing
- Integration testing
- Testing object oriented applications
- Introduction to system testing
- *Introduction* to testing distributed/web applications
- *Introduction* to testing service oriented applications
- Code inspection, review techniques, and walkthroughs
- Alpha, Beta, and Validation testing

- Software configuration management (version control and build management)
- Bug life cycle and bug management
- Test planning, documentation, and tracking progress

Grading Policies*:*

Homework/Projects - 45%
Quizzes/Lecture Review Activities- 15%
Midterm I – 12.5%
Midterm II – 12.5%
Final Exam - 15%

Final Grade	Percentage
A+	>= 97%
A	>= 90% and < 97%
B+	>= 87% and < 90%
B	>= 80% and < 87%
C+	>= 77 and < 80%
C	>= 70% and < 77%
D	>= 60% and < 70%
	< 60%

Homework/Project Objectives: Apply testing, configuration management and quality control techniques in software development process. Use appropriate tools in testing, test documentation, and configuration management.

Readings, Special Materials, Required Activities:

Text Book (Required)

[1] Software Testing: A Craftsman's Approach, Fourth Edition, Paul C. Jorgensen, *Grand Valley State University, Allendale, Michigan, USA*, 2013, ISBN: 978-1466560680, CRC press.

Other References (Few copies will be available in the library reference section):

1. Testing Computer Software, 2nd Edition, Cem Kaner, Jack Falk, Hung Q. Nguyen, ISBN 9780471358466.

2. How We Test Software at Microsoft, Alan Page; Ken Johnston; Bj Rollison, 9780735624252
3. Selenium WebDriver Practical Guide, Satya Avasarala, ISBN: 9781782168867, January 24, 2014
4. Practical Unit Testing with JUnit and Mockito, Tomek Kaczanowski, ISBN: 978-8393489398, April 23, 2013

Tools-Required

- git, and git-hub - Version Control
- Ecplise, JUnit, MockitO, eclEmma- Testing Java applications
- soapUI - Testing Service Oriented Applications
- Open Office Bug Tracking System – Bug Management
- Selenium and associated tools – Testing client server web applications
- Sonorqube – code review and quality analysis tool

Topics and Lecture Hours Distribution-*Tentative*

Total Number of Weeks: 16

Topics:

1. **Introduction to Software Quality and Quality Assurance Tasks – 1 Week**
   - Software quality
   - Elements of software quality assurance
   - SQA tasks, goals, and metrics
   - Software quality standards (ISO, IEEE …etc)
   - Software quality assurance throughout software life cycle
   - Testing vs. Software quality assurance

2. **Testing Background – 1 Week**
   - Software testing : A bigger picture
   - Testing levels
     - Verification and Validation
     - Testing Strategies
   - Testing matrices

- How much testing is enough?
- Testing throughout the software life cycle
- What is a test case?
- Anatomy of a test case
- Managing test cases

3. Functional Testing (Black-Box testing) – 2 Weeks
   - Equivalent partition
   - Boundary value analysis
   - Use-case based testing

4. Bug Management – I Week
   - Effective bug reporting
   - The bug workflow
   - Tracking bugs

5. Model-based testing - I Week
   - Testing with models?
   - Finite state machine model testing

6. Structure Based Testing Techniques (White-Box Testing) – 2 Weeks
   - Introduction
   - Path coverage and McCabe Cyclomatic complexity
   - Control structure coverage
   - Combinations of conditions
   - Data path testing
   - Automated unit testing and Test-driven development

7. Testing Object Oriented Applications - 2 Weeks
   - Design of object oriented applications and Testability
   - Test-cases and class hierarchy
   - Use case based test design
   - State based testing
   - Inter-class testing

   7.1 Code Matrices and Testability of a Program
   - Introduction
   - Refactoring and regression testing

8. Introduction to Configuration Management - 0.5 Weeks
   - Version control and build management

9. Integration Testing - 1 Week
   - Various approaches used for integration testing including, Incremental, Top-Down, Bottom-Up, Smoke, Sandwich, and Regression testing

10. Introduction to System Testing – 0.5 Weeks
    - Various types of systems tests including Recovery testing, Security testing, Performance testing, and Deployment testing

11. Testing Web/Distributed (including SOA) Applications – 2.5 Weeks
    - Design considerations in distributed applications
    - Introduction to distributed software architectures

    7.1 Testing Web Applications
    - Quality dimensions of web applications
    - Errors within web application environments
    - User interface testing
    - Content Testing
    - Component testing
    - Performance Testing
    - Configuration testing

    7.2 Testing SOA applications
    - Testing services
    - Deployment testing
    - Integration Testing
    - Model-based SOA application testing using petri-net modeling

8 Requirement Based Testing Techniques - 0.5 Weeks
- Validation : Alpha, Beta , and Acceptance testing

9 Software Quality Control Beyond Testing – 1.5 Weeks
- Inspections, Reviews, and Walkthroughs
- Defects prevention and process improvements
- Comparison of quality assurance techniques

Attendance: You are expected to attend the class regularly and be punctual in class. You are responsible for all the announcements that are made in the class whether you attend or not. If you wish to be withdrawn

from the class, you should complete the withdrawal form. DO NOT JUST STOP COMING TO CLASS! Submitting a completed withdraw form to the registrar office is the only guaranteed way to officially withdrawing from the class.

Grading Appeals : Any questions or appeals on grades of homework, projects, or quizzes must be done in writing by completing the "Grade Inquiry Form" within a week from the day the assignment was returned or comments were published on-line. State the problem and the rationale for any change in grade in your appeal.

Cooperation and Academic Honesty: You are encouraged to cooperate in study group on preparing homework, projects, quizzes and exams. However, anything you turn in must be your own work: You must write up your own solution with your own understanding. If you use an idea that is found in a book or other sources, or that was developed by someone else or jointly with some group, make sure you acknowledge the source and/or the names of the persons in the write-up for each problem. The instructor and the TAs will CAREFULLY check any possible proliferation or plagiarism. We may also use the software tools to check any assignment that you submitted for grading. The Department of Computer Science and Engineering expects all students to adhere to ASU's policy on Academic Dishonesty. These policies can be found in the Code of Student Conduct
http://www.asu.edu/studentaffairs/studentlife/judicial/academic_integrity.htm
ALL cases of cheating or plagiarism will be handed to the Dean's office. Penalties include a failing grade in the class, a note on your official transcript that shows you were punished for cheating, suspension, expulsion and revocation of already awarded degrees.

Announcement: Official announcements will be made either in the class or in the course web page. Make sure you regularly (at least once a day every two days) check the web page for any announcement. Announcements made in the class supersede the announcements posted in the course web page/printed material.

Absence & Make-Up Policies : Accommodations will be made for religious observances provided that students notify the instructor at the beginning of the semester concerning those dates. Students who expect to miss class due to officially university-sanctioned activities should inform the instructor early in the semester. Alternative arrangements will generally be made for any examinations and other graded in-class work affected by such absences. The preceding policies are based on ACD 304–04, "Accommodation for Religious Practices" and ACD 304–02, "Missed Classes Due to University-Sanctioned Activities." Also, of the student is having medical and other health related issues, make-ups are possible provided the student brings a valid doctor's note. No late homework/lab/quiz/lecture review activity submissions will be accepted for any other reason than mentioned above.

Classroom Behavior: Cell phones and pagers (must be/or state alternative rule) turned off during class to avoid causing distractions. The use of recording devices is not permitted during class. Any violent or

threatening conduct by an ASU student in this class will be reported to the ASU Police Department and the Office of the Dean of Students.

Disability Accommodations: Suitable accommodations will be made for students having disabilities and students should notify the instructor as early as possible if they will require same. Such students must be registered with the Disability Resource Center and provide documentation to that effect.

Sexual Discrimination: Title IX is a federal law that provides that no person be excluded on the basis of sex from participation in, be denied benefits of, or be subjected to discrimination under any education program or activity. Both Title IX and university policy make clear that sexual violence and harassment based on sex is prohibited. An individual who believes they have been subjected to sexual violence or harassed on the basis of sex can seek support, including counseling and academic support, from the university. If you or someone you know has been harassed on the basis of sex or sexually assaulted, you can find information and resources at https://sexualviolenceprevention.asu.edu/faqs.

As a mandated reporter, I am obligated to report any information I become aware of regarding alleged acts of sexual discrimination, including sexual violence and dating violence. ASU Counseling Services, https://eoss.asu.edu/counseling, is available if you wish discuss any concerns confidentially and privately.

Disclaimer: I reserve the right to revise this syllabus as necessary. Any changes to the syllabus will be announced via the course website.