King Abdullah II School for Information Technology
Department of Computer Information Systems
Fall 2013

Software Verification and Validation (1902715)

Instructor: **Dr. Omar Adwan**
Office: **210**
Phone: **22610**
Email: **adwanoy@gmail.com**
Class time: **Wed. 3:30-6:30**
Class Room: **KASIT 203**
Office Hours:

Course Description

Verification and Validation course provides all aspects of the system engineering discipline employing a rigorous methodology for evaluating and assessing the correctness and quality of system and software throughout the system/software life cycle. It includes the following topics: Planning verification and validation; verification and validation techniques. Software testing: Component testing, integration testing and system testing. Testing techniques; Verification and formal methods. Critical systems validation. Transition Systems, Kripke Structure, Linear Temporal Logic (LTL), Computational Tree Logic (CTL), Binary Decision Diagrams, Modeling Real-Time Systems, Algorithms for Verifying Real Time Systems.

Course Objectives

This course is designed for advanced graduate students. The course objectives are to provide students with an:

- Understanding of the concepts and theory related to software testing, verification and validation methods and techniques throughout the software life cycle.
- Develop software testing plans and models
- Apply different types of testing approaches
- Perform software inspection and program analysis
- Develop and apply continuous verification methods
- Conduct test reviews and formalize documentation
- Validate critical systems

Topics

This course covers the following topics:

- Introduction to Software Development Process
- Overview of Software Verification Activities
- Overview of Software Validation Activities
- Fundamentals of Software Testing and Analysis
- Basic Software Testing Techniques

- Software Testing Problems and Methods
- Software Testing Process

Course Outline

Overview of Software Process

- Improvement Models
- SEI CMM
- ISO SPICE
- Bootsttrap
- ISO-9000-3
- TickIT
- Trillium

Software Development Process

- Overview of Software Development Lifecycle
- Models
- Waterfall
- Prototyping
- Spiral

Economic Justification for Software V&V

- Software Defect Cost Modelv Find/Fix Cycle Costs
- Time-to-Market vs. Quality Tradeoff

Software Verification Overview

- The Inspection Process
- Applying the Inspection Process
- Requirements Inspections
- Design Inspections
- Code Inspections
- Test Inspections
- Practice Code Inspection

In-Process Measurements

- Measurement Attributes
- IEEE Software Metrics Framework
- Overview of In-process Measures
- Project Estimation Measures
- Complexity Measures
- Quality Measures
- Product Measures
- Process Measures

Configuration Management Overview

- Identification
- Building a Software Parts List
- Building a Software Parts Diagram
- Baseline Management

- Baseline Change Assessment
- Auditing and Reporting

Software Validation Overview

- Testing is Hard
- Levels of Testing
- Methods of Testing
- Goals and Objectives of Testing
- Unit Testing
- Integration Testing
- Validation Testing

Software Validation Testing

- Test Planning
- Test Estimation and Development
- Test Execution

How do you know when to start?

- Software Validation Readiness Review

How do you know when you're done?

- Completion Criteria

Measurements

- Time Measures
- Thoroughness Measures
- Quality Measures
- Tools

Software Reliability Growth

- Modeling Approaches
- Model Assumptions
- Model Selection Process
- Tools

Achieving Quality Software On Time

- Scheduling Forwards
- Estimating and Scheduling using the Yellow
- Sticky Method

Evaluation

Midterm Exam	30%
Project	15%
Assignments	15%
Final Exam	40%

Assignments

The Assignments include research paper critiques. For each paper, students should write a review answering each of the following questions:

1. What problems(with prior work or the lack thereof) were addressed or surveyed by authors?
2. What solutions were proposed or surveyed by authors?
3. What are the technical strengths and main contributions of the paper's proposed solution?
4. What are the technical weaknesses of the paper's proposed solution?
5. What suggestions do you have to improve upon the paper's ideas?

Project

Students are to submit a paper-like on any topic related to V&V. A published paper is a plus with a full mark on project.

Text Book

Marcus S. Fisher, Software Verification and Validation: An Engineering and Scientific Approach, Springer, 2007. ISBN: 0387327258.

References

1. Steven R. Rakitin, Artech, Software Verification and Validation for Practitioners and Managers, 2nd Edition, Computer Science Library, 2003, ISBN: 1580532969
2. Srinivasan Desikan, and Gopalaswamy Ramesh, Software Testing: Principles and Practices, Pearson Education, 2006, ISBN: 817758295
3. Mauro Pezzè, and Michael Young,Software Testing and Analysis: Process, Principles, and Techniques, Wiley, 2007, ISBN10: 0471455938
4. Ian Sommerville. Software Engineering, 8th Edition. Addison Wesley, 2006.
5. J.M. Spivey. The Z Notation: A Reference Manual. 2nd edition, Prentice-Hall, 1992.
6. Roger S. Pressman. *Software Engineering A practitioner's Approach, Sixth Edition.* McGraw-Hill, 2004.
7. Shari Lawrence Pfleeger, *Software Engineering: Theory and Practice*, 2nd Ed., Prentice-Hall, 2001.
8. University of Jordan E-library: http://e-library/
9. IEEE Transactions on software engineering
10. ACM Transactions on software engineering