SOFTWARE TESTING

CO3015 / CO5252

Course goals

Provide students with

- basic knowledge of software testing, including roles and importance of software testing, testing processes and planning, software testing and technical review techniques;
- and help students to develop abilities to inspect software as well as design and test software through practical activities.

Course description

- Introduction to Software testing, including role and the importance of software testing, testing processes, testing level and testing techniques overview.
- Software technical review.
- White-box testing techniques, including control-flow testing and data-flow testing.
- Black-box testing techniques, including equivalence class partitioning, boundary value analysis, decision table, cause-effect graph, pairwise, use-case techniques.
- Unit testing, including test case design, incremental testing, top-down and bottom-up testing, and test automation.
- Test-driven development
- Functional testing, system testing, regression testing and acceptance testing.
- And other testing issues.

Textbook / References

- [1] Glenford J. Myers, Corey Sandler, and Tom Badgett. 2011. The Art of Software Testing (3rd. ed.). Wiley Publishing.
- [2] Andreas Spillner, Tilo Linz, Hans Schaefer. 2014. Software Testing Foundations: A Study Guide for the Certified Tester Exam (4th ed.). Rocky Nook Computing.
- **[3] Introduction to Software Testing (ed. 2), Ammann and Offutt, 2017, Cambridge University Press.**
- **[4] Jorgensen, Paul C. 2013. Software Testing: a Craftsman's Approach (4th ed.). Software Testing. Hoboken: CRC Press.**
- [5] Pezzè, M., & Young, M. (2008). Software testing and analysis: Process, principles, and techniques. Hoboken, N.J.: Wiley.
- [6] Lee Copeland. 2003. A Practitioner's Guide to Software Test Design. Artech House, Inc., USA.
- [7] Dorothy Graham, Erik Van Veenendaal, Isabel Evans, and Rex Black. 2008. Foundations of Software Testing: ISTQB Certification. Intl Thomson Business Pr.
- [8] Ilene Burnstein. 2010. Practical Software Testing: A Process-Oriented Approach (1st. ed.). Springer Publishing Company, Incorporated.

Learning Outcome

- L.O.1. Explain the role of software testing, the types of software testing and the implementation process
  - L.O.1.1. Explain the roles and importance of software testing
  - L.O.1.2. Explain the software testing types and levels
  - L.O.1.3. Explain the software testing processes and planning
- L.O.2. Use techniques to test software
  - L.O.2.1. Use the checklist-based technique in software technical review
  - L.O.2.2. Use the white-box testing technique to test the software
  - L.O.2.3. Use the back-box testing technique to test the software
- L.O.3. Understand test-driven software development and test automation
  - L.O.3.1. Understand the test-driven development process
  - L.O.3.2. Understand the automating testing

Evaluation

- Quizzes / In-class activities: 10%
- Midterm: 10%
- Project: 30%
- Final exam: 50%

- ***Additional grading policy: if any of those grading components is less than 3.0 / 10.0, the total mark of the course will be the minimum of them.***

Quizzes & Project

- Quizzes:
  - Every week, BKeL
- Project = 03 mini-projects:
  - Proj#1 (10%): Code Review
  - Proj#2 (10%): Black-box testing
  - Proj#3 (10%): Web automated testing

Assessment Plan

Q&I: Quizzes / In-class activities
Proj#x: project
M: Midterm exam
F: Final exam

L.O	Assessment
L.O.1. Explain the role of software testing, the types of software testing and the implementation process	
L.O.1.1. Explain the roles and importance of software testing	Q&I, M
L.O.1.2. Explain the software testing types and levels	Q&I, M
L.O.1.3. Explain the software testing processes and planning	Q&I, M, F
L.O.2. Use techniques to test software	
L.O.2.1. Use the checklist-based technique in software technical review	Q&I, Proj#1
L.O.2.2. Use the white-box testing technique to test the software	Q&I, M, F
L.O.2.3. Use the back-box testing technique to test the software	Q&I, Proj#2, F
L.O.3. Understand test-driven software development and test automation	
L.O.3.1. Understand the test-driven development process	Q&I, F
L.O.3.2. Understand the automating testing	Q&I, Proj#3

Teaching plan

W	Date	Topics	L.O.	Notes
1		Course outline Ch1. Introduction	L.O.1	
2		Ch2. Software Technical Review	L.O.2.1	Team up
3		Ch3. White-box testing	L.O.2.2	
4		Ch3. White-box testing (cont.)	L.O.2.2	
5		Ch4. Black-box testing (cont.)	L.O.2.2	Proj#1
6		Ch4. Black-box testing (cont.)	L.O.2.2	
7		Midterm / Discussion		
		Midterm break		
8		Ch4. Black-box testing	L.O.2.3	
9		Ch4. Black-box testing (cont.)	L.O.2.3	
10		Ch4. Black-box testing (cont.)	L.O.2.3	
11		Ch4. Black-box testing (cont.)	L.O.2.3	
12		Ch5. Unit testing	L.O.1.3	Proj#2
13		Ch6. Other testing activities	L.O.1.3	
14		Ch6. Test-driven development process	L.O.3	
15		Ch7. Other testing issues	L.O.1.3	Proj#3

Contact

- Assoc. Prof. Bùi Hoài Thắng
- Through BKeL
- Email: bhthang@hcmut.edu.vn
  - Email subject: begin with [CO3015]