MONASH University
Information Technology

FIT4004
System validation and verification, quality and standards

Unit Guide

Semester 1, 2013

The information contained in this unit guide is correct at time of publication. The University has the right to change any of the elements contained in this document at any time.

Last updated: 04 Mar 2013

Table of Contents

FIT4004 System validation and verification, quality and standards - Semester 1, 2013

This unit covers the fundamental products, processes and techniques for system validation and verifications including testing methodologies, static program analysis and code quality measurement and monitoring. Open-source tools will be used to apply in practice knowledge learnt about software testing from a theoretical perspective. Inspection and testing methodologies, analysis of artefacts, robustness, performance analysis configuration management, quality assurance plan and standards, compliance, assessment, certification issues are covered. It shows how to predict, analyse and control defects in complex software systems.

Mode of Delivery

Clayton (Day)

Contact Hours

2 hrs lectures/wk, 1 hr tutorial/wk

Workload requirements

Estimated weekly commitment needed for the unit, including classes, reading, assessment, time needed for computer access, and other activities:

- two-hour lecture
- one-hour tutorial
- one-hour unsupervised lab/tute activity in the MUSE Lab to get familiarised with tools, work on assignments, self study, etc.
- a minimum of 2-3 hours of personal study per one hour of contact time in order to satisfy the reading and assignment expectations.
- You will need to allocate up to 5 hours per week in some weeks, for use of a computer, including reading research papers for an assignment and lab discussions with class members.

Unit Relationships

Prohibitions

CSE4431

Prerequisites

FIT2004, FIT2024, FIT3042, FIT3077 and one of FIT2002 or FIT3086
or students must be enrolled in FIT Masters program at Monash

Chief Examiner

Dr Yuan-Fang Li

Campus Lecturer

Clayton

Yuan-Fang Li

Tutors

Clayton

Kefeng Xuan

Academic Overview

Learning Outcomes

At the completion of this unit students will have -A knowledge and understanding of:

- the role of validation and verification methods in the system life cycle;
- key issues in software testing, testing levels and testing activities;
- testing techniques - based on testers experience - adhoc testing, exploratory testing - specification-based - equivalence partioning, boundary-value analysis, finite-state machine based, random testing - code-based - control-flow and data-flow technique - fault-based - error seeding, mutation testing - usage-based - reliability measures, operational profile - based on type of apps - web based, OO, component testing - selection and combination of techniques;
- test related measures - evaluation of software under test - fault density, types of faults - evaluation of tests done - criteria such as coverage, thoroughness; mutation score;
- empirical work, replication experiments vs case study.

Developed attitudes that enable them to:

- adhere to software quality engineering principles;
- recognise the importance of adhering to software engineering principles of Validation and Verification and standards in the design and development of test methods;
- have an understanding of inspection and debugging approaches, configuration management, performance, and quality standards issues.

Developed the skills to:

- use IDEs such as Eclipse, NetBeans and IntelliJ IDEA and unit testing with JUnit, build management tool such as Maven, continuous integration tool such as Hudson, and code quality monitoring tools such as Sonar and Cobertura, and other similar products to help detect software system defects;
- conduct continuous integration process for the application at unit, integration & system testing level with access to SVN, Hudson Continuous Integration (CI) server etc;
- appreciate how assertion mechanisms impact reasoning;
- be able to analyse and control defects in complex systems.

Unit Schedule

Week	Activities	Assessment
0		No formal assessment or activities are undertaken in week 0
1	Overview, Fundamentals of Software Testing	
2	Mathematics for software testing & quality: set theory, graph theory, etc.	Weekly assessed tutorials start in Week 2
3	Black-box testing	Unit, Integration, System and Continuous testing - Phase 1 due Week 3
4	White-box testing I	
5	White-box testing II	
6	Component testing	
7	Software quality & metrics	Unit, Integration, System and Continuous testing - Phase 2 due Week 7
8	System testing	
9	Object-oriented testing	
10	Mutation testing	
11	A comparison of software verification and validation techniques	Unit, Integration, System and Continuous testing - Phase 3 due Week 11
12	Web testing overview	
	SWOT VAC	No formal assessment is undertaken in SWOT VAC
	Examination period	LINK to Assessment Policy: http://policy.monash.edu.au/policy-bank/academic/education/assessment/assessment-in-coursework-policy.html

*Unit Schedule details will be maintained and communicated to you via your learning system.

Assessment Summary

Examination (2 hours): 50%; In-semester assessment: 50%

Assessment Task	Value	Due Date
Unit, Integration, System and Continuous testing - Phase 1	10%	Week 3
Unit, Integration, System and Continuous testing - Phase 2	15%	Week 7
Unit, Integration, System and Continuous testing - Phase 3	15%	Week 11
Tutorial Participation	10%	Weekly
Examination 1	50%	To be advised

Teaching Approach

Lecture and tutorials or problem classes

This teaching and learning approach provides facilitated learning, practical exploration and peer learning.

Assessment Requirements

Assessment Policy

Faculty Policy - Unit Assessment Hurdles
(http://www.infotech.monash.edu.au/resources/staff/edgov/policies/assessment-examinations/unit-assessment-hu

Academic Integrity - Please see the Demystifying Citing and Referencing tutorial at
http://lib.monash.edu/tutorials/citing/

Assessment Tasks

Participation

Students are expected to attend all tutorials and participate in active discussions. Tutorials will carry 10% assessment and will be assessed on participation.

- **Assessment task 1**

  Title:
  Unit, Integration, System and Continuous testing - Phase 1

  Description:
  The first phase of the semester-long project consists of two parts:

  ♦ System setup,
  ♦ Implementation as specified in the assignment specification, and
  ♦ Unit testing of the various component classes in the code base provided to and extended by you.

  Weighting:
  10%

  Criteria for assessment:
  This assignment is evaluated on the correctness and completeness of the work:

  ♦ System setup, and

  ◊ Successful setup of the working environment on personal laptops, and
  ◊ Successful connection to the integration server.

  ♦ Implementation

  ◊ Completeness of the implementation.

  ♦ Unit testing

  ◊ Developing functionality according to a given specification,
  ◊ Successfully invoking automated builds on the server, and
  ◊ Development of non-trivial unit tests appropriate for specified system functionality.

  No written or file submission is required for this assessment. It will be based only on a demo in the tutorial and answering queries & during an interview.

  Due date:
  Week 3

- **Assessment task 2**

Title:

Unit, Integration, System and Continuous testing - Phase 2

Description:

The second phase of the semester-long project focuses on integration testing.

You will continue to develop the system based on a given specification and write unit tests and integration tests for the newly developed components. Such tests will also need to be automatically executed on the continuous integration server.

Students enrolled in FIT5171 will be given additional creative tasks to complete in this assignment. Details will be given in the assignment description.

Weighting:

15%

Criteria for assessment:

The assignment will be assessed by its correctness and completeness.

- ♦ Sufficient functionality in the different layers of the system.
- ♦ Sufficient testing adequacy of the developed functionality.

No written or file submission is required for this assessment. It will be based only on a demo in the tutorial and answering queries & during an interview.

Due date:

Week 7

- **Assessment task 3**

Title:

Unit, Integration, System and Continuous testing - Phase 3

Description:

The third phase of the semester-long project focuses on system testing.

You will continue to develop the system based on a given specification and write unit tests, integration tests and system tests for the newly developed components. Such tests will also need to be automatically executed on the continuous integration server.

Students enrolled in FIT5171 will be given additional creative tasks to complete in this assignment. Details will be given in the assignment description.

Weighting:

15%

Criteria for assessment:

The assignment will be assessed by its correctness and completeness.

- ♦ Sufficient functionality of the whole system (backend and the Web frontend).
- ♦ Sufficient testing adequacy of the components and the system.

Assessment will consist of a demo & interview in the tutorial and a short written report.

Due date:

Week 11

- **Assessment task 4**

  Title:
  Tutorial Participation
  Description:
  Tutorials run from Week 2 through to Week 12. Questions related to lectures, assignment demos and literature reading will be conducted in tutorials.
  Weighting:
  10%
  Criteria for assessment:
  Active participation in discussions.
  Due date:
  Weekly

Examinations

- **Examination 1**

  Weighting:
  50%
  Length:
  2 hours
  Type (open/closed book):
  Open book
  Electronic devices allowed in the exam:
  None

Learning resources

Reading list

- Jorgensen, Paul C. (2008), **Software Testing, A Craftsman's Approach**, 3rd Edition, Auerbach Publications.
- M Pezze and M Young (2007), **Software Testing and Analysis**, Wiley Publ.
- J F Peters and W Pedrycz (2000), **Software Engineering: An Engineering Approach**, J Wiley Publ.
- Robert V. Binder (1999), **Testing Object-Oriented Systems: Models, Patterns, and Tools**, Addison-Wesley.
- David A Sykes John D McGregor (2001), **Practical Guide to Testing Object-Oriented Software**, Addison-Wesley.
- Daniel J.Mosley, Bruce A. Posey (2002), **Just Enough Software Test Automation**, Addison-Wesley.
- Jerry Gao, H S Tsao and Ye Wu (2003), **Testing and Quality Assurance for Component-based Software**, Artech House.
- Matt Staats, Michael W. Whalen, Mats P.E. Heimdahl. **Programs, Tests, and Oracles: The Foundations of Testing Revisited**. In *Proceedings of the 33rd International Conference on Software Engineering (ICSE'11)*. 2011, pages 391-400. ACM.
- Fraser, G., Wotawa, F., & Ammann, P. E. (2009). **Testing with model checkers: a survey**. *Software Testing, Verification and Reliability*, *19*(3), 215-261.
- Majumdar, R., & Sen, K. (2007, May). **Hybrid concolic testing**. In *Software Engineering, 2007. ICSE 2007. 29th International Conference on* (pp. 416-426). IEEE.

- Jia, Y., & Harman, M. (2011). **An analysis and survey of the development of mutation testing**. *Software Engineering, IEEE Transactions on*, *37*(5), 649-678.
- Arcuri, A., Iqbal, M. Z., & Briand, L. (2010). **Formal analysis of the effectiveness and predictability of random testing**. In *Proceedings of the 19th international symposium on Software testing and analysis* (pp. 219-230). ACM.

Monash Library Unit Reading List
http://readinglists.lib.monash.edu/index.html

Feedback to you

Types of feedback you can expect to receive in this unit are:

- Informal feedback on progress in labs/tutes
- Graded assignments without comments
- Solutions to tutes, labs and assignments

Extensions and penalties

Submission must be made by the due date otherwise penalties will be enforced.

You must negotiate any extensions formally with your campus unit leader via the in-semester special consideration process: http://www.infotech.monash.edu.au/resources/student/equity/special-consideration.html.

Returning assignments

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Assignment submission

It is a University requirement (http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-procedures.html) for students to submit an assignment coversheet for each assessment item. Faculty Assignment coversheets can be found at http://www.infotech.monash.edu.au/resources/student/forms/. Please check with your Lecturer on the submission method for your assignment coversheet (e.g. attach a file to the online assignment submission, hand-in a hard copy, or use an online quiz).

Online submission

If Electronic Submission has been approved for your unit, please submit your work via the learning system for this unit, which you can access via links in the my.monash portal.

Required Resources

Please check with your lecturer before purchasing any Required Resources. Limited copies of prescribed texts are available for you to borrow in the library, and prescribed software is available in student labs.

The MUSE Lab in Bldg 26/G13 is the lab used for this unit. However, students are required to install the required software on their laptops to minimise interruptions to development and testing. The required software include:

- Tools for Software testing such as JUnit 4.x (latest vers)
- Java build management Apache Maven 2.x or 3.x
- The version control system Subversion
- An open source or commercial Java IDE (Eclipse, NetBeans or IntelliJ IDEA)

Software may be:

- Downloaded from:
    - http://www.eclipse.org/downloads/
    - http://www.jetbrains.com/idea/download/
    - http://tortoisesvn.net/downloads.html
    - http://maven.apache.org/download.html
- A free classroom license for IntelliJ IDEA will be provided via Moodle

Recommended Resources

Students are encouraged to use their own laptops to work on project assignments. All required software can be downloaded and installed onto personal laptops.

Recommended text(s)

Jorgensen, Paul C. (2008). *Software Testing, A Craftsman's Approach*. (3rd Edition) Auerbach Publications.

Other Information

Policies

Monash has educational policies, procedures and guidelines, which are designed to ensure that staff and students are aware of the University's academic standards, and to provide advice on how they might uphold them. You can find Monash's Education Policies at: www.policy.monash.edu.au/policy-bank/academic/education/index.html

Key educational policies include:

- Plagiarism; http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-policy.html
- Assessment in Coursework Programs; http://www.policy.monash.edu/policy-bank/academic/education/assessment/assessment-in-coursework-po
- Special Consideration; http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.ht
- Grading Scale; http://www.policy.monash.edu/policy-bank/academic/education/assessment/grading-scale-policy.html
- Discipline: Student Policy; http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-discipline-policy.html
- Academic Calendar and Semesters; http://www.monash.edu.au/students/dates/
- Orientation and Transition; http://intranet.monash.edu.au/infotech/resources/students/orientation/
- Academic and Administrative Complaints and Grievances Policy; http://www.policy.monash.edu/policy-bank/academic/education/management/complaints-grievance-policy.h
- Code of Practice for Teaching and Learning; http://www.policy.monash.edu.au/policy-bank/academic/education/conduct/suppdocs/code-of-practice-teac

Graduate Attributes Policy

http://www.policy.monash.edu/policy-bank/academic/education/management/monash-graduate-attributes-policy.h

Student services

The University provides many different kinds of support services for you. Contact your tutor if you need advice and see the range of services available at http://www.monash.edu.au/students. For Sunway see http://www.monash.edu.my/Student-services, and for South Africa see http://www.monash.ac.za/current/.

Monash University Library

The Monash University Library provides a range of services, resources and programs that enable you to save time and be more effective in your learning and research. Go to www.lib.monash.edu.au or the library tab in my.monash portal for more information. At Sunway, visit the Library and Learning Commons at http://www.lib.monash.edu.my/. At South Africa visit http://www.lib.monash.ac.za/.

Disability Liaison Unit

Students who have a disability or medical condition are welcome to contact the Disability Liaison Unit to discuss academic support services. Disability Liaison Officers (DLOs) visit all Victorian campuses on a regular basis.

Website: http://www.monash.edu/equity-diversity/disability/index.htmlTelephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Commuity Services at 03 55146018 at SunwayEmail: dlu@monash.eduDrop In: Equity and Diversity Centre, Level 1, Building 55, Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Sunway Campus

Your feedback to Us

Monash is committed to excellence in education and regularly seeks feedback from students, employers and staff. One of the key formal ways students have to provide feedback is through the Student Evaluation of Teaching and Units (SETU) survey. The University's student evaluation policy requires that every unit is evaluated each year. Students are strongly encouraged to complete the surveys. The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

For more information on Monash's educational strategy, see:

www.monash.edu.au/about/monash-directions and on student evaluations, see: www.policy.monash.edu/policy-bank/academic/education/quality/student-evaluation-policy.html

Previous Student Evaluations of this Unit

Previous feedback has highlighted the following strengths in this unit:

- Practical, progmamming based project
- Challenging and stimulating tutorials and assignment

Student feedback has also informed improvements to this unit including:

- The assignment timing and weights have been adjusted to reflect their relative difficulty level
- Practical sessions have been introduced to make tools/systems more accessible to students unfamiliar with them

If you wish to view how previous students rated this unit, please go to https://emuapps.monash.edu.au/unitevaluations/index.jsp