FIT5171
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

**FIT5171 System validation and verification, quality and standards - Semester 1, 2013**

- Mode of Delivery .......................................................... 1
- Contact Hours ............................................................... 1
- Workload requirements .................................................. 1
- Unit Relationships .......................................................... 1
  - Prohibitions .............................................................. 1
  - Prerequisites ............................................................. 1
- Chief Examiner ............................................................. 1
- Campus Lecturer ........................................................... 2
  - Clayton ................................................................. 2
- Tutors ............................................................................ 2
  - Clayton ................................................................. 2

## Academic Overview ....................................................... 3
  - Learning Outcomes ...................................................... 3

## Unit Schedule ............................................................... 4
  - Assessment Summary .................................................. 4
  - Teaching Approach ..................................................... 5

## Assessment Requirements ............................................ 6
  - Assessment Policy ....................................................... 6
  - Assessment Tasks ....................................................... 6
  - Participation ............................................................. 6
  - Examinations ............................................................ 8
    - Examination 1 ......................................................... 8
  - Learning resources .................................................... 8
  - Reading list ............................................................. 8
  - Feedback to you ........................................................ 9
  - Extensions and penalties ............................................ 9
  - Returning assignments ............................................... 9
  - Assignment submission ............................................ 9
  - Online submission ................................................... 9
  - Required Resources .................................................. 9
  - Recommended Resources ........................................... 10
    - Recommended text(s) ............................................... 10

## Other Information ...................................................... 11
  - Policies ....................................................................... 11
    - Graduate Attributes Policy ..................................... 11
  - Student services ........................................................ 11
  - Monash University Library ......................................... 11
  - Disability Liaison Unit ............................................... 12
  - Your feedback to Us .................................................. 12
  - Previous Student Evaluations of this Unit ...................... 12
FIT5171 System validation and verification, quality and standards - Semester 1, 2013

This unit covers the core software engineering disciplines concerned with managing and delivering quality software. Topics include processes, tools and techniques for system validation and verification, including major commercial tools used in industry. It shows how to predict, analyse and control defects in complex software systems. Inspection and testing methodologies, analysis of artefacts, robustness, performance analysis configuration management, quality assurance plan and standards including ISO9000/AS39000, compliance, assessment, certification issues are covered.

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, FIT4004

Prerequisites

Recommended knowledge: programming in C, C++ and Java; OOSE, Analysis, Design and Programming; OO Method - UML notation, method and SE process; Project Management.

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 knowledge and understanding of the role of validation & verification methods in the system life cycle;
- have gained practical experience in using commercial validation tools to help detect software system defects;
- appreciate how assertion mechanisms impact reasoning;
- be able to analyse and control defects in complex systems;
- have an understanding of inspection & testing methods, configuration management, performance, and quality standards issues.
### Unit Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Activities</th>
<th>Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>No formal assessment or activities are undertaken in week 0</td>
</tr>
<tr>
<td>1</td>
<td>Overview, Fundamentals of Software Testing</td>
<td>Weekly assessed tutorials start in Week 2</td>
</tr>
<tr>
<td>2</td>
<td>Mathematics for software testing &amp; quality: set theory, graph theory, etc.</td>
<td>Unit, Integration, System and Continuous testing - Phase 1 due Week 3</td>
</tr>
<tr>
<td>3</td>
<td>Black-box testing</td>
<td>Unit, Integration, System and Continuous testing - Phase 1 due Week 3</td>
</tr>
<tr>
<td>4</td>
<td>White-box testing I</td>
<td><strong>Unit Schedule details will be maintained and communicated to you via your learning system.</strong></td>
</tr>
<tr>
<td>5</td>
<td>White-box testing II</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Component testing</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Software quality &amp; metrics</td>
<td>Unit, Integration, System and Continuous testing - Phase 2 due Week 7</td>
</tr>
<tr>
<td>8</td>
<td>System testing</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Object-oriented testing</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Mutation testing</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>A comparison of software verification and validation techniques</td>
<td>Unit, Integration, System and Continuous testing - Phase 3 due Week 11</td>
</tr>
<tr>
<td>12</td>
<td>Web testing overview</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SWOT VAC</td>
<td>No formal assessment is undertaken in SWOT VAC</td>
</tr>
</tbody>
</table>

### Assessment Summary

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

<table>
<thead>
<tr>
<th>Assessment Task</th>
<th>Value</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit, Integration, System and Continuous testing - Phase 1</td>
<td>10%</td>
<td>Week 3</td>
</tr>
<tr>
<td>Unit, Integration, System and Continuous testing - Phase 2</td>
<td>15%</td>
<td>Week 7</td>
</tr>
<tr>
<td>Unit, Integration, System and Continuous testing - Phase 3</td>
<td>15%</td>
<td>Week 11</td>
</tr>
<tr>
<td>Tutorial Participation</td>
<td>10%</td>
<td>Weekly</td>
</tr>
<tr>
<td>Examination 1</td>
<td>50%</td>
<td>To be advised</td>
</tr>
</tbody>
</table>
Unit Schedule

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

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 Requirements

• 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 Requirements

• 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

• M Pezze and M Young (2007), Software Testing and Analysis, Wiley Publ.
• Daniel J.Mosley, Bruce A. Posey (2002), Just Enough Software Test Automation, Addison-Wesley.
Assessment Requirements


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.


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.
Assessment Requirements

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.jetbrains.com/idea/download/
  - http://tortoisesvn.net/downloads.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)**

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
- Special Consideration; http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.html
- 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/

Graduate Attributes Policy

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

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.html
Telephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Community Services at 03 55146018 at Sunway
Email: dlu@monash.edu
Drop 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, programming 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