MONASH University
Information Technology

ENG1003
Engineering mobile apps

Unit Guide

Semester 1, 2015

Last updated: 18 Feb 2015

Table of Contents

ENG1003 Engineering mobile apps - Semester 1, 2015

This unit introduces students to the use of Information Technology (IT) in modern engineering practice. Students will learn an object-oriented approach to both computer systems and software engineering for solving engineering problems.

Students will work in small teams to develop a mobile application that meets a contemporary need in engineering. The fundamental stages in the software development lifecycle will be introduced, including requirements analysis, design, implementing and testing. Students will use IT tools to support the engineering process.

Mode of Delivery

- Clayton (Day)
- Malaysia (Day)

Workload Requirements

2 hours lectures, 3 hours of laboratory/workshop activities and 7 hours of private study per week.

See also Unit timetable information

Chief Examiner

Dr Kris Ryan

Campus Lecturer

Clayton

Dr. Michael Wybrow

Consultation hours: TBA

Malaysia

Dr. Esyin Chew

Consultation hours: TBA

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

Academic Overview

Learning Outcomes

On successful completion of this unit students should be able to:

1. describe, with examples, the multidisciplinary nature of modern engineering problems
2. describe the interaction between developments in IT and their use in modern engineering practice
3. create and evaluate a simple multi-platform object-oriented architecture design
4. design and implement a mobile application that utilises the device capabilities to solve an engineering problem
5. design and perform simple black box acceptance testing
6. use IT tools, including an Integrated Development Environment (IDE), shared code repository and version control, bug-tracking and design drawing tools, for aspects of the software development process
7. interpret and produce written technical documentation in standard design formalism
8. complete tasks as part of a team, and communicate effectively with team members
9. communicate effectively with clients as part of the software development process

Unit Schedule

Week	Activities	Assessment
0	Pre-reading for week 1. Complete Belbin survey. See http://eng1003.eng.monash.edu/	No formal assessment or activities are undertaken in week 0
1	Quiz. Lectures: Introduction, Software and Software Engineering. 3-hour lab class. Pre-reading for week 2.	Quiz.
2	Quiz. Lectures: Introduction to web apps, JavaScript programming 1. 3-hour lab class Pre-reading for week 3.	Quiz.
3	Quiz. Lectures: Mobile hardware architecture, JavaScript programming 2. 3-hour lab class Pre-reading for week 4.	Quiz.
4	Quiz. Lectures: Software engineering process, JavaScript programming 3. 3-hour lab class Pre-reading for week 5.	Quiz.
5	Quiz. Lectures: Requirements analysis, Web app user interfaces. 3-hour lab class. Pre-reading for week 6.	Quiz. Assignment 1 due.
6	Quiz. Lectures: Version control and configuration management, Debugging JavaScript. 3-hour lab class. Pre-reading for week 7.	Quiz. Assignment 1 presentation (in lab time).
7	Quiz. Lectures: Communication and project management. App persistence. 3-hour lab class. Pre-reading for week 8.	Quiz.
8	Quiz. Lectures: Software design, Algorithms and data structures 1, App persistence. 3-hour lab class. Pre-reading for week 9.	Quiz. Lab-based architectural design activity.
9	Quiz. Lectures: Quality and Risk Management, Algorithms and data structures 2. 3-hour lab class. Pre-reading for week 10.	Quiz.
10	Quiz. Lectures: Software testing, Network communication. 3-hour lab class. Pre-reading for week 11.	Quiz. Lab-based acceptance testing exercise.
11	Quiz. Lecture: Mobile UI and Interaction Design, Security and privacy. 3-hour lab class. Pre-reading for week 12.	Quiz.
12	Quiz. Lecture: Performance and responsiveness. Guest lecture on app development. 3-hour lab class.	Quiz. Assignment 2 team presentation (in lab time). Assignment 2 due.
	SWOT VAC	No formal assessment is undertaken during 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.

Teaching Approach

Other

The teaching approach is flipped classroom. Students will be introduced to topics via online ePub and video material. Lectures will be expert-led sessions where concepts are demonstrated interactively with the class. Students will gain practical experience via hands-on lab classes.

Assessment Summary

Continuous assessment: 60%
Examination (3 hours): 40%
Students are required to achieve at least 45% in the total continuous assessment component and at least 45% in the final examination component and an overall mark of 50% to achieve a pass grade in the unit. Students failing to achieve this requirement will be given a maximum of 45% in the unit.

Assessment Task	Value	Due Date
Weekly pre-lecture quizzes	12%	Weekly, before delivery of the first Workshop (lecture) each week.
Assignment 1: Height measurement app	10%	End of week 5. Team presentation during week 6 lab class.
Assignment 2: Walking route tracking app	30%	End of week 12. Team presentation during week 12 lab class.
Lab work assessments	8%	During lab classes in weeks 8 and 10.
Examination 1	40%	To be advised

Assessment Requirements

Assessment Policy

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

Academic Integrity - Please see resources and tutorials at
http://www.monash.edu/library/skills/resources/tutorials/academic-integrity/

Assessment Tasks

Participation

- **Assessment task 1**

  Title:
  Weekly pre-lecture quizzes
  Description:
  You will complete a total of twelve online weekly quizzes. They will test that students have read and comprehended the week's preparatory ePub and video material.
  Weighting:
  12%
  Criteria for assessment:

  - Comprehension of weekly preparatory materials

  Due date:
  Weekly, before delivery of the first Workshop (lecture) each week.

- **Assessment task 2**

  Title:
  Assignment 1: Height measurement app
  Description:
  As a member of your assigned project team, you will create a simple mobile app that uses the device sensors to estimate the height of a real-world object. Source code to your app will be submitted as a team. (7%)

  Your team will deliver a 10 minute oral presentation (in lab class) describing and demonstrating your app and detailing any issues you encountered. (3%)
  Weighting:
  10%
  Criteria for assessment:

  - Correctness of the produced app
  - Quality of app source code, including code documentation
  - Clarity and quality of oral presentation

  You and your team member will receive a combined mark, except:

  - Your presentation will be accessed individually.

  Due date:

End of week 5. Team presentation during week 6 lab class.

- **Assessment task 3**

Title:

Assignment 2: Walking route tracking app

Description:

You will work together as a team to produce a GPS-based mobile app for recording, measuring and visualising walking paths. This assignment consists of several component assessment items with the following associated marks:

- App code and functionality — 16% (group)
- Use of appropriate tools — 5% (individual)
- Mock client oral presentation (in lab) — 5% (individual+group)
- Production of external technical documentation — 4% (group)

You will be provided with initial skeleton code for the app. This skeleton will provide pre-written code that implement map-based canvas with basic interactivity, as well as an API for storing user data locally and remotely.

You will create an app that polls the GPS at regular intervals and reads information about the user's location. You will implement functionality to display location and accuracy information on a map. You will extend your app to store historical position information into a route structure and produce code to measure the distance of such routes. Your team will submit source code to your app at this stage by tagging a version of the code in your version control repository. (8%)

You will further extend the code of your app to persist user route data and provide an interface for viewing previously recorded routes as well as recording new ones. You will implement functionality that simplifies routes to remove unnecessary points and outliers. You will further extend the app to check validity of routes, remove unnecessary detours on a route, and display simplified routes visually on the map. Your team will submit source code to your app at this stage by tagging a version of the code in your version control repository. (8%)

You will be tested on your use of appropriate tools for software development. You will use a source code editor and debugger for writing code. You will use a shared version control system for managing revisions of the app source and handling commits by multiple team members. You will use a online software for document management, team communication, project management and issue tracking. (5%)

Your team will deliver a 15 minute oral presentation (in lab class) describing and demonstrating your app and detailing any issues you encountered. Team presentations will be made during lab time, with each student presenting for 4-5 minutes. (5%)

Your team will produce external technical documentation for your app including external design documentation and a short guide/reference manual. Teams will be assessed based on the quality of these documents. (4%)

Weighting:

30%

Criteria for assessment:

- Correct functionality and behaviour of the produced app
- Quality of app source code, including code documentation

- Individual clarity and quality of oral presentation
- Structure, appropriateness, and level of team-client presentation
- Participation and appropriate use of tools for team software development
- Comprehensiveness of technical documentation

You and your team member will receive a combined mark, except:

- The history of your interaction and use of the version control repository, team communication system and shared documents will be examined and assessed individually.
- You will be assessed on your individual presentation style, as well as the team presentation as a whole.

Due date:

End of week 12. Team presentation during week 12 lab class.

- **Assessment task 4**

Title:

Lab work assessments

Description:

You will complete several lab-based assessment tasks. These tasks are each to be related to the project work from Assignment 2 but do not depend on the team's progress and will be completed and submitted during lab class time.

These tasks will be:

- Evaluate and revise an object-oriented architecture design — 4% (group)
- Perform and document the results of acceptance testing — 4% (group)

In the first task, you will be given a set of requirements for an expanded version of the assignment 2 app as well as a rough OO architectural design. In your project teams, you will evaluate the existing architecture and revise it to suit the given requirements. As a team you will submit documents detailing your updated design, including artefacts such as class diagrams and sequence diagrams. (4%)

In the second task, you will design and conduct black box acceptance testing of a pre-provided software component. In your teams, you will write and submit a short report detailing your testing approach and the results. (4%)

Weighting:

8%

Criteria for assessment:

- Understanding of design issues and quality of revised architectural design
- Adequacy of testing coverage and quality of the associated report

You and your team member will receive a combined mark.

Due date:

During lab classes in weeks 8 and 10.

Examinations

- **Examination 1**

Weighting:
40%
Length:
3 hours
Type (open/closed book):
Open book
Electronic devices allowed in the exam:
None

Learning resources

Monash Library Unit Reading List (if applicable to the unit)
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 with comments
- Graded assignments without comments
- Quiz results

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.monash.edu.au/exams/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/student-academic-integrity-managing-pla 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 electronic submission). Please note that it is your responsibility to retain copies of your assessments.

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.

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

Faculty resources and policies

Important student resources including Faculty policies are located at http://intranet.monash.edu.au/infotech/resources/students/

Graduate Attributes Policy

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

Student Charter

www.opq.monash.edu.au/ep/student-charter/monash-university-student-charter.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 Malaysia 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 Malaysia, 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 Commuity Services at 03 55146018 at Malaysia
- 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, Malaysia Campus

Other

Engineers Australia Stage 1 competencies

Engineers Australia Accreditation Policy of Professional Engineering Programs requires that programs demonstrate how engineering graduates are prepared for entry to the profession and achieve Stage 1 competencies. The following information describes how this unit contributes to your development of these competencies. (Note: not all competencies may be emphasised in this unit).

Stage 1 competency	How the compency is developed in this unit
1. Knowledge and Skills base	
1.1. **Comprehension, theory based understanding** of the underpinning natural and physical sciences and the engineering fundamentals applicable to the engineering discipline.	This unit gives an introduction to the theory underpinning programming and software engineering. Several of the lab exercise are designed to reinforce aspects of the theory.
1.2. **Conceptual understanding** of the mathematics, numerical analysis, statistics, and computer and information sciences, which underpin the engineering discipline.	This unit describes the computer science and mathematics underpinning computer hardware architecture, computation and basic algorithm analysis. Students will complete lab exercises exploring these topics.
1.3. **In-depth understanding** of specialist bodies of knowledge within the engineering discipline.	Not covered in this unit—the unit only introduces specialist concepts at a high level.
1.4. **Discernment** of knowledge development and research directions within the engineering discipline.	Not covered in this unit.
1.5. **Knowledge** of engineering design practice and contextual factors impacting the engineering discipline.	The unit introduces students to various software engineering practices, including requirements analysis, software design, communication, project management, risk management, and software testing. Students utilise this knowledge in labs and assignment work.
1.6. **Understanding** of the scope, principles, norms, accountabilities and bounds of sustainable engineering practice in the specific discipline.	Not covered in this unit.
2. Engineering application ability	
2.1. **Application** of established engineering methods to complex engineering problem solving.	The unit presents software engineering process models and practices for effective software engineering. It also discusses the appropriateness of varous computer science approaches for different tasks, though only at a high level.
2.2 **Fluent application** of engineering techniques, tools and resources.	This unit teaches systematic software engineering techniques and tools, specifically those for version control and configuration management, debugging, and team communication.
2.3. **Application** of systematic engineering synthesis and design processes.	The unit presents key aspects of the software design process and has students participate in high-level software architectural design through lab exercises.
2.4. **Application** of systematic approaches to the	This unit teaches systematic software

conduct and management of engineering projects.	engineering processes for the effective conduct and management of software development projects, especially related to project management, risk management and communication.
3. Professional and personal attributes	
3.1. **Ethical** conduct and professional accountability.	Not covered in this unit.
3.2. **Effective** oral and written communication in professional and lay domains.	Students work in small teams for much of the lab and assignment work, requiring substantial communication. They give two assessed oral presentation, the second beign a mock client presentation.
3.3. **Creative**, innovative and proactive demeanour.	Not covered in this unit.
3.4. **Professional** use and management of information.	Students are reuired to manage information during the assignements and submit these documents as part of the assessment. Information to be used: software code, design documents, team communications, user manuals, issue and task lists.
3.5. **Orderly** management of self, and professional conduct.	This is covered by the various teamwork focused activities in the lab classes and assignments.
3.6. **Effective** team membership and team leadership.	Team roles are examined, reflected upon and used for team allocation. All assignemnts and a portion of the lab work is performed in teams.

Relationship between Unit Learning Outcomes and BSE Course Outcomes

No.	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7	C0 8	CO 9	CO 10	CO 11	CO 12	CO 13
1		X				X	X	X					
2	X	X				X	X						
3			X		X	X		X					
4	X				X				X	X	X		
5			X		X								
6			X		X					X	X		
7			X		X	X		X					
8								X	X	X	X		
9								X					

Relationship between Unit Learning Outcomes and Assessments

No.	Quizzes	Assignments	Lab exercies	Exam
1	X			X
2	X			X
3		X	X	X

Other Information

4	X		
5		X	
6	X		
7	X		X
8	X		
9	X		