ITEC800

Systems Engineering Process

S1 Evening 2014

Computing

Contents

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff
Unit Convenor
Carl Svensson
carl.svensson@mq.edu.au
Contact via carl.svensson@mq.edu.au
Organise via email

Credit points
4

Prerequisites
COMP355 or COMP365 or ISYS355

Corequisites

Co-badged status

Unit description
Topics covered in this unit include the software development life cycle and its phases, generic and project-specific aspects of the life cycle, risk, estimation methods, process models and modelling, process maturity, process improvement, metrics, experimentation, reliability, experience packaging, organisational issues, socio-technical aspects of process, software evolution, process-centred development environments and standards.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at https://www.mq.edu.au/study/calendar-of-dates

Learning Outcomes

On successful completion of this unit, you will be able to:

Evaluate software system development processes and the context in larger systems projects.

Evaluate both requirements and design process activities in developing software systems.

Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.

Evaluate verification, validation and testing procedures in developing software systems.

Evaluate project management processes and quality processes in developing software systems.

Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Assessment Tasks

Name	Weighting	Due
Fortnightly Submissions	20%	Fortnightly
Project Deliverable 1	5%	Saturday Week 3
Project Deliverable 2	10%	Saturday Week 6
Project Progress Presentation	5%	Saturday Week 9
Project Deliverable 3	10%	Saturday Week 10
Project Deliverable 4	10%	Saturday Week 13
Final examination	40%	End of Semester

Fortnightly Submissions

Due: **Fortnightly**
Weighting: **20%**

The individually marked practical and small research submissions assess the understanding of fortnightly topics as we go along in the semester.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Project Deliverable 1

Due: **Saturday Week 3**

Weighting: **5%**

The group project proposal and project plan (not exceeding a total of 20 A4 sides) including the problem to be tackled, programming language (s) being used, scope, schedule, project resources, group organisation, tracking mechanisms and control mechanisms for the project.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate project management processes and quality processes in developing software systems.

Project Deliverable 2

Due: **Saturday Week 6**
Weighting: **10%**

This assignment gets students to develop two documents. The first is the document communicating findings from their requirements analysis; the second is the test specification that details how they would expect the system to be tested and what types of tests it would have to pass in order to be deemed acceptable. The assignment is a group based assignment with individual contributions clearly outlined and graded accordingly.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.

Project Progress Presentation

Due: **Saturday Week 9**
Weighting: **5%**

This is a group interim presentation assessed with individual grades. Students give a presentation outlining the results of their analysis of the problem and intent for the design of the system. It is an opportunity for students to assess other presentations on their content, presentation style and coverage of the problem.

This will be a way for individuals to see the insight of other groups after having completed the majority their own work, and provide some feedback for their next deliverable. It also provides a way to evaluate presentation styles and give students insight into different approaches taken so they can prepare for their final presentation at the end of the semester.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Project Deliverable 3

Due: **Saturday Week 10**
Weighting: **10%**

This assignment gets students to develop the design of their system that addresses the requirements outlined in the first half of the semester. The document being submitted is a system design document that includes: user interface layout concepts, program navigation, data definitions, analysis and design class diagrams, selected sequence diagrams, selected use cases, selected state diagrams, a requirements traceability matrix and a list of any important assumptions. The assignment is a group based assignment with individual contributions clearly outlined and graded accordingly.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Project Deliverable 4

Due: **Saturday Week 13**
Weighting: **10%**

The components of this deliverable include:

- a revised project plan based on what occured during the semester
- user documentation for the system

- a presentation of the sofware functionality, software non-functional qualities, the software processes used and future work

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.
- Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Final examination

Due: **End of Semester**
Weighting: **40%**

The exam will be a written exam with no multiple choice sections. It will be held in the usual examination period of the semester. Students have 3 hours written time plus 10 minutes reading time to complete the two sections of the exam.

On successful completion you will be able to:

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.
- Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Delivery and Resources

The main textbook for the 2014 offering of ITEC800

1. (SE): **Software Engineering**, *Somerville. I*, Latest Edition (9th Edition) *Pearson Education Limited*, ISBN:137 035 152;

(The full set of slides for this textbook are available from Sommerville's site)

Supplementary books (highly recommended but not compulsory for students)

While the unit will be drawing from the prescribed textbook listed above, the resources listed below will also be used to supplement some of the material. The list (of some great resources) below is provided for information purposes in case students would like to refer to them, but it is **NOT** compulsory for students. Where available, copies have been placed in the reserve section of the Macquarie University Library so that all students have access to the below supplementary resources.

1. (BF): **Systems Engineering and Analysis**, *Blanchard. B. S, Fabrycky. W. J,* Latest Edition (5th Edition, 2011), *Pearson.*
2. (RIDEC): **Refactoring: Improving the Design of Existing Code**, *Fowler. M, and others*, Latest Edition (1999, printed 2008), *Addison Wesley.*

Classes:

Classes for ITEC800 this semester run on selected Saturdays (Weeks 1, 3, 5, 7, 9, 11 and 13) throughout the semester. Please refer to the Unit Schedule for details. In week 1 we will be running a 3 hour lecture block from 9am till noon.

In weeks 3, 5, 7, 9, 11 and 13 we will be running a 3 hours lecture block from 9am and then a 3 hours lab session from 1pm.

Be prepared to participate in discussions, ask and answer questions, and provide perspectives from your own background and workplaces. Lectures will be informative but also exploratory in nature, and so it is certainly recommended that a high participation and attendance level be maintained throughout the course.

Other resources:

Lecture material, assignment specifications and extra readings not listed above will be provided through the iLearn web page for ITEC800.

Submission methods for assessment tasks:

All soft copy assignment submissions / marks will be done through the ITEC800 page on iLearn. This will be the official form of assignment submission accepted for this course unless otherwise stated in the specification or through the discussion process of an extension (see below).

Late submission:

Late submission of individual work will incur a 20% penalty for every 24 hours it is late. So within 24 hours, the maximum mark that can be obtained is 80% of the full grade for that assessment task; between 24 and 48 hours, the maximum mark that can be obtained is 60% of the full grade; and so on. No extra documentation is required unless the student does wishes to have an extension (see below) applied.

Late submissions of group based assignments are not permitted unless under exceptional circumstances with documentary evidence provided to the unit convenor which may include medical certificates as per the Department of Computing policy. One person being sick does not mean the group cannot submit work. Students are recommended to have a backup plan for group based submissions.

Extensions:

Extensions without a grade penalty may be provided to groups or individuals who contact the unit convenor and can provide documentary evidence of illness or other misadventure. If approved, a new submission timeline and submission method will be discussed on a case by case basis.

Students are strongly advised to contact the unit convenor as early as possible if there are any issues that will not make an on-time submission possible.

Exam:

The final exam will focus on content covered in the classes throughout the semester. Please see the assessments section for details on the final exam.

Website and access to unit material:

The web page and content for this unit can be found at iLearn: https://ilearn.mq.edu.au/login/MQ/. Note that the unit content is not publicly available and requires for you to log in to access.

What has changed from last year:

This year we have added some individual tasks which make up a 20% component of this course. A 5% presentation on project progress has been added, and the final exam has been taken to 40% of the workload.

Unit Schedule

Classes for ITEC800 in 2014 Session 1 are scheduled to run on Saturdays (odd semester weeks) during the semester. The first Saturday is one 3 hour block. The weeks following week 1 are two block sessions. The timetable below is divided into the weeks where classes are scheduled. Please refer to the announcements section in iLearn for any details regarding topic, schedule or room alterations.

Week 1 Saturday 8th March	A systems perspective, socio-technical considerations and Agile. Requirements: Elicitation, analysis, specification and modelling.
Week 2	No Class
Week 3 Saturday 22nd March	System Evaluation I: Validation, verification and testing. System Evaluation II: Quality and security (considerations and management). **Week 1 prac & research tasks due**
Week 4	No Class
Week 5 Saturday 5th April	System Design I: System partitioning, architectures, requirements allocation and traceability. System Design II: Components, design patterns and reuse. **Week 3 prac & research tasks due**
Week 6	No Class
Week 7 Saturday 3rd May	System Design III: Distributed systems, embedded systems, mission-critical systems, "big data" systems. System Design IV: System design for operational feasibility (including metrics and considerations for usability, maintainability, reliability and supportability). **Week 5 prac & research tasks due**
Week 8	No Class
Week 9 Saturday 17th May	System implementation, integration and operation: Rapid software development, CBSE, software evolution. **Student Presentations** **Week 7 prac & research tasks due**
Week 10	No Class
Week 11 Saturday 31st May	Systems Management I: People and teams, project management, risk, estimation, resourcing, organisational issues and logistics. Systems Management II: Change management, configuration management, process improvement. **Week 9 prac & research tasks due**
Week 12	No Class

Week 13 Saturday 14th June	Advanced treatment of system life-cycles: System development life-cycles, product life-cycles, process models, software evolution. **Student Presentations** **Week 11 prac & research tasks due**
Exam Period	**Final Exam**

Learning and Teaching Activities

Readings

Pre-lecture recommended readings to assist learning concepts and stimulate discussion topics.

Lectures

Fortnightly discussion of key concepts.

Individual practical and research assignments

Individual tasks and research activities focusing on selected concepts covered in lectures.

Group-based assignments

Working in small groups to discuss and generate documentation for different phases in the software development process.

Student presentations

Group student presentations given by each individual member of the group outlining different aspects investigated in their group-based assignment. Feedback on content and presenation style is included as part of the student presentation process.

Assignment feedback

Academic evaluation of assignments.

Student presentation feedback

Students and staff assessing student presentation modes and styles of content delivery providing written feedback.

Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central. Students should be aware of the following policies in particular with regard to Learning and Teaching:

Academic Honesty Policy http://mq.edu.au/policy/docs/academic_honesty/policy.html

Assessment Policy http://mq.edu.au/policy/docs/assessment/policy.html

Grading Policy http://mq.edu.au/policy/docs/grading/policy.html

Grade Appeal Policy http://mq.edu.au/policy/docs/gradeappeal/policy.html

Grievance Management Policy http://mq.edu.au/policy/docs/grievance_management/policy.html

Disruption to Studies Policy http://www.mq.edu.au/policy/docs/disruption_studies/policy.html *The Disruption to Studies Policy is effective from March 3 2014 and replaces the Special Consideration Policy.*

In addition, a number of other policies can be found in the Learning and Teaching Category of Policy Central.

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: https://students.mq.edu.au/support/student_conduct/

Student Support

Macquarie University provides a range of support services for students. For details, visit http://students.mq.edu.au/support/

Learning Skills

Learning Skills (mq.edu.au/learningskills) provides academic writing resources and study strategies to improve your marks and take control of your study.

- Workshops
- StudyWise
- Academic Integrity Module for Students
- Ask a Learning Adviser

Student Services and Support

Students with a disability are encouraged to contact the Disability Service who can provide appropriate help with any issues that arise during their studies.

Student Enquiries

For all student enquiries, visit Student Connect at ask.mq.edu.au

IT Help

For help with University computer systems and technology, visit http://informatics.mq.edu.au/help/.

When using the University's IT, you must adhere to the Acceptable Use Policy. The policy applies to all who connect to the MQ network including students.

Graduate Capabilities

PG - Discipline Knowledge and Skills

Our postgraduates will be able to demonstrate a significantly enhanced depth and breadth of knowledge, scholarly understanding, and specific subject content knowledge in their chosen fields.

This graduate capability is supported by:

Learning outcomes

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Assessment tasks

- Fortnightly Submissions
- Project Deliverable 1
- Project Deliverable 2
- Project Progress Presentation
- Project Deliverable 3
- Project Deliverable 4
- Final examination

Learning and teaching activities

- Pre-lecture recommended readings to assist learning concepts and stimulate discussion topics.
- Fortnightly discussion of key concepts.
- Individual tasks and research activities focusing on selected concepts covered in lectures.

PG - Critical, Analytical and Integrative Thinking

Our postgraduates will be capable of utilising and reflecting on prior knowledge and experience, of applying higher level critical thinking skills, and of integrating and synthesising learning and knowledge from a range of sources and environments. A characteristic of this form of thinking is the generation of new, professionally oriented knowledge through personal or group-based critique of practice and theory.

This graduate capability is supported by:

Learning outcomes

- Evaluate software system development processes and the context in larger systems projects.
- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Assessment tasks

- Project Deliverable 1
- Project Deliverable 2
- Project Progress Presentation
- Project Deliverable 3
- Project Deliverable 4
- Final examination

Learning and teaching activities

- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Working in small groups to discuss and generate documentation for different phases in the software development process.

PG - Research and Problem Solving Capability

Our postgraduates will be capable of systematic enquiry; able to use research skills to create new knowledge that can be applied to real world issues, or contribute to a field of study or practice to enhance society. They will be capable of creative questioning, problem finding and problem solving.

This graduate capability is supported by:

Learning outcome

- Evaluate both requirements and design process activities in developing software systems.

Assessment tasks

- Fortnightly Submissions
- Project Deliverable 2
- Project Deliverable 3

Learning and teaching activities

- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Working in small groups to discuss and generate documentation for different phases in the software development process.

PG - Effective Communication

Our postgraduates will be able to communicate effectively and convey their views to different social, cultural, and professional audiences. They will be able to use a variety of technologically supported media to communicate with empathy using a range of written, spoken or visual formats.

This graduate capability is supported by:

Learning outcomes

- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.
- Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Assessment tasks

- Fortnightly Submissions
- Project Deliverable 1
- Project Deliverable 2
- Project Progress Presentation
- Project Deliverable 3
- Project Deliverable 4

Learning and teaching activities

- Working in small groups to discuss and generate documentation for different phases in the software development process.
- Group student presentations given by each individual member of the group outlining different aspects investigated in their group-based assignment. Feedback on content and presenation style is included as part of the student presentation process.
- Students and staff assessing student presentation modes and styles of content delivery providing written feedback.

PG - Engaged and Responsible, Active and Ethical Citizens

Our postgraduates will be ethically aware and capable of confident transformative action in relation to their professional responsibilities and the wider community. They will have a sense of connectedness with others and country and have a sense of mutual obligation. They will be able to appreciate the impact of their professional roles for social justice and inclusion related to national and global issues

This graduate capability is supported by:

Learning outcomes

- Evaluate both requirements and design process activities in developing software systems.
- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Assessment tasks

- Project Deliverable 1
- Project Deliverable 2
- Project Deliverable 3

Learning and teaching activities

- Working in small groups to discuss and generate documentation for different phases in the software development process.
- Students and staff assessing student presentation modes and styles of content delivery providing written feedback.

PG - Capable of Professional and Personal Judgment and

Initiative

Our postgraduates will demonstrate a high standard of discernment and common sense in their professional and personal judgment. They will have the ability to make informed choices and decisions that reflect both the nature of their professional work and their personal perspectives.

This graduate capability is supported by:

Learning outcomes

- Analyse operational feasibility considerations such as usability, maintainability, reliability and security in developing software systems.
- Evaluate verification, validation and testing procedures in developing software systems.
- Evaluate project management processes and quality processes in developing software systems.

Assessment tasks

- Fortnightly Submissions
- Project Deliverable 1
- Project Deliverable 2
- Project Deliverable 3
- Project Deliverable 4

Learning and teaching activities

- Group student presentations given by each individual member of the group outlining different aspects investigated in their group-based assignment. Feedback on content and presenation style is included as part of the student presentation process.
- Students and staff assessing student presentation modes and styles of content delivery providing written feedback.

Standards

Four standards, namely HD, D, CR, P summarize as many different levels of achievement. Each standard is precisely defined to help students know what kind of performance is expected to deserve a certain mark. The standards corresponding to the learning outcomes of this unit are given below:

Standards Matrix

	Unsatisfactory level of demonstrating creativity, project leadership, strong analytical and design skills. Unsatisfactory level of demonstrating ability to assist in solving real-world software	Sometimes demonstrate creativity, project leadership, strong analytical and design skills. Sometimes demonstrate ability to assist in solving real-world software development problems and manage project risk.	Often demonstrate creativity, project leadership, strong analytical and design skills. Demonstrate ability to assist in solving real-world software development problems and manage project risk.	Demonstrate creativity, project leadership, strong analytical and design skills. Demonstrate ability to assist in solving real-world software development problems and manage project risk.	Demonstrate creativity, project leadership, strong analytical and design skills. Demonstrate ability to solve real-world software development problems and manage project risk.
	Unsatisfactory level of applying techniques and knowledge to develop requirements and design deliverables software systems to a very high standard of quality.	Sometimes apply techniques and knowledge to develop requirements and design deliverables software systems to a very high standard of quality.	Often apply techniques and knowledge to develop requirements and design deliverables software systems to a very high standard of quality.	Frequently apply techniques and knowledge to develop requirements and design deliverables software systems to a very high standard of quality.	Always apply techniques and knowledge to develop requirements and design deliverables software systems to a very high standard of quality.
	Unsatisfactory level of applying techniques and knowledge to assess operational feasibility considerations in software systems to a very high standard of quality.	Sometimes apply techniques and knowledge to assess operational feasibility considerations in software systems to a very high standard of quality.	Often apply techniques and knowledge to assess operational feasibility considerations in software systems to a very high standard of quality.	Frequently apply techniques and knowledge to assess operational feasibility considerations in software systems to a very high standard of quality.	Always apply techniques and knowledge to assess operational feasibility considerations in software systems to a very high standard of quality.

	Unsatisfactory level of applying techniques and knowledge to implement and test software systems to a very high standard of quality.	Sometimes apply techniques and knowledge to implement and test software systems to a very high standard of quality.	Often apply techniques and knowledge to implement and test software systems to a very high standard of quality.	Frequently apply techniques and knowledge to implement and test software systems to a very high standard of quality.	Always apply techniques and knowledge to implement and test software systems to a very high standard of quality.
	Unsatisfactory level of applying techniques and knowledge to conduct project management processes for software systems to a very high standard of quality.	Sometimes apply techniques and knowledge to conduct project management processes for software systems to a very high standard of quality.	Often apply techniques and knowledge to conduct project management processes for software systems to a very high standard of quality.	Frequently apply techniques and knowledge to conduct project management processes for software systems to a very high standard of quality.	Always apply techniques and knowledge to conduct project management processes for software systems to a very high standard of quality.
	Unsatisfactory level of demonstrating an ability to communicate in a structured form. Unsatisfactory level of assessing others in their communication attempts.	Sometimes apply an ability to communicate in a structured form. Sometimes able to assess others in their communication attempts.	Often apply an ability to communicate in a structured form. Often able to assess others in their communication attempts.	Frequently apply an ability to communicate in a structured form. Frequently able to assess others in their communication attempts.	Always apply an ability to communicate in a structured form. Always able to assess others in their communication attempts.

Grading

At the end of the semester, you will receive a grade that reflects your achievement in the unit

Fail (F): does not provide evidence of attainment of all learning outcomes. There is missing or partial or superficial or faulty understanding and application of the fundamental concepts in the field of study; and incomplete, confusing or lacking communication of ideas in ways that give little attention to the conventions of the discipline.

Pass (P): provides sufficient evidence of the achievement of learning outcomes. There is demonstration of understanding and application of fundamental concepts of the field of study; and communication of information and ideas adequately in terms of the conventions of the discipline. The learning attainment is considered satisfactory or adequate or competent or capable in relation to the specified outcomes.

Credit (Cr): provides evidence of learning that goes beyond replication of content knowledge or skills relevant to the learning outcomes. There is demonstration of substantial understanding of fundamental concepts in the field of study and the ability to apply these concepts in a variety of contexts; plus communication of ideas fluently and clearly in terms of the conventions of the discipline.

Distinction (D): provides evidence of integration and evaluation of critical ideas, principles and theories, distinctive insight and ability in applying relevant skills and concepts in relation to learning outcomes. There is demonstration of frequent originality in defining and analysing issues or problems and providing solutions; and the use of means of communication appropriate to the discipline and the audience.

High Distinction (HD): provides consistent evidence of deep and critical understanding in relation to the learning outcomes. There is substantial originality and insight in identifying, generating and communicating competing arguments, perspectives or problem solving approaches; critical evaluation of problems, their solutions and their implications; creativity in application.

In this unit, your final grade depends on your performance in each part of the assessment. For each task, you receive a mark that combines your standard of performance regarding each learning outcome assessed by this task. Then the different component marks are added up to determine your total mark out of 100. Your grade then depends on this total mark and your overall standards of performance.

In order to pass the unit, you must

- Have performed satisfactorily in the internal (assessment) components of the course.
- Have satisfactory performance in the final examination.
- This means that you may fail the unit if you do not submit satisfactory submissions for the assignments OR do not perform satisfactorily in the exam.
- Department of Computing expectations are that students have to perform satisfactorily in the final exam as well as in their internal work/assignments.

In order to get a higher grade in the unit, you must

- Obtain the required total number of marks (Credit - 65, Distinction - 75, High Distinction - 85).
- Meet the criteria outlined in the Grading section above
- Perform at the equivalent band in the standards matrix.

Changes since First Published

Date	Description
25/02/2014	Updated "standards" section to reflect the change in wording of the learning outcomes