



ITEC800

Systems Engineering Process

S1 Evening 2015

Dept of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>Assessment Tasks</u>	3
<u>Delivery and Resources</u>	7
<u>Unit Schedule</u>	8
<u>Learning and Teaching Activities</u>	10
<u>Policies and Procedures</u>	10
<u>Graduate Capabilities</u>	11
<u>Changes from Previous Offering</u>	17
<u>Grading and Standards</u>	17

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 Carl Svensson carl.svensson@mq.edu.au Appointment by 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 quality 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
<u>Prac and Research Tasks</u>	30%	Friday 8pm: Weeks 3, 5, 9 & 13
<u>Group Project: Deliverable 1</u>	5%	Friday Week 3
<u>Group Project: Deliverable 2</u>	10%	Friday Week 6
<u>Project Progress Presentation</u>	5%	Saturday Week 9 (In Class)
<u>Group Project: Deliverable 3</u>	10%	Friday Week 10
<u>Group Project: Deliverable 4</u>	10%	Week 13 (see description)
<u>Final Exam</u>	30%	Exam Period

Prac and Research Tasks

Due: **Friday 8pm: Weeks 3, 5, 9 & 13**

Weighting: **30%**

4 x Individual submissions assessing the understanding of topics covered in the course. Each submission has both practical and research tasks associated with the deliverable.

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

Group Project: Deliverable 1

Due: **Friday Week 3**

Weighting: **5%**

One group submission outlining the proposal and project plan.

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.

Group Project: Deliverable 2

Due: **Friday 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 quality software systems.

Project Progress Presentation

Due: **Saturday Week 9 (In Class)**

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 is intended to be a way for individuals to see the insight of other groups after having completed the majority their own work, and for them to 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 verification, validation and testing procedures in developing quality software systems.
- Understand how to communicate and assess communications for a targeted audience in both written and spoken forms.

Group Project: Deliverable 3

Due: **Friday 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. 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 quality software systems.

Group Project: Deliverable 4

Due: **Week 13 (see description)**

Weighting: **10%**

The components of this deliverable include:

- a revised project plan based on what occurred during the semester (Due Friday)
- user documentation for the system (Due Friday)
- an (in class) presentation of the software 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 quality 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 Exam

Due: **Exam Period**

Weighting: **30%**

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 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 quality 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

Where available, limited numbers of hard-copy versions of textbooks listed below have been placed in the reserve section of the Macquarie University Library.

The primary textbook used for the 2015 offering of ITEC800

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

(The full set of companion slides for this textbook are available from [Somerville'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.

1. (BF): **Systems Engineering and Analysis**, *Blanchard. B. S, Fabrycky. W. J*, Latest Edition (5th Edition, 2011), *Pearson*.
2. (RIDECE): **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 for up to four hours on Saturdays (Weeks 1, 2, 3, 4, 5, 8, 9, 10, 11, 12 and 13) from 9:00am until 1:00pm. Please refer to the Unit Schedule for details. The breakdown of each scheduled Saturday has a lecture (from 9:00am to 11:00am) and a mixed session (from 11:00am to 1:00pm).

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.

Note: Due to the Easter weekend public holiday, there will be no classes scheduled for ITEC800 in week 6 (Saturday April 4).

Note: Due to the ANZAC DAY Public Holiday, there will be no classes scheduled for ITEC800 in week 7 (Saturday April 25).

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 hour period it is late. So within 24 hours after the submission deadline, 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 wishes to have an extension (see below) with no penalty 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 urged to have backup plans 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 a login to access the content and discussions.

Unit Schedule

Unit Schedule (week by week breakdown)

Class Schedule	Topics - (Subject to change)
Week 1 Saturday 28th February	Taking a systems perspective - Introduction and consolidation, identifying the system and environment, socio-technical considerations, ethics and professionalism, initial plan-based and Agile considerations.

Class Schedule	Topics - (Subject to change)
Week 2 Saturday 7th March	Systems analysis - Problem analysis and requirements engineering.
Week 3 Saturday 14th March	System evaluation and testing - System Evaluation I: Validation, verification and testing.
Week 4 Saturday 21st March	System evaluation and testing - System Evaluation II: Quality and security.
Week 5 Saturday 28th March	System design - System architectures, system partitioning and requirements allocation, traceability, design approaches and documentation considerations.
Week 6	<i>No Class for ITEC800 in this week due to classes falling on the Easter Weekend Public Holiday.</i>
Teaching Break	<i>Two week mid-semester teaching break</i>
Week 7	<i>No Class for ITEC800 in this week due to classes falling on ANZAC Day.</i>
Week 8 Saturday 2nd May	System design - Design considerations for systems: Distributed systems, embedded systems, mission-critical systems, "big data" systems.
Week 9 Saturday 9th May	System design - Targeting system design for operational feasibility (including metrics and considerations for usability, maintainability, reliability and supportability).
Week 10 Saturday 16th May	System deployment and maintenance - System implementation, integration and operation, reuse opportunities, Component Based Software Engineering, system evolution and decommissioning.
Week 11 Saturday 23rd May	Systems management - Change and configuration management, process improvement.
Week 12 Saturday 30th May	Systems management - People and teams, project management, risk, estimation, resourcing, organisational issues and Agile.

Class Schedule	Topics - (Subject to change)
Week 13	Advanced treatment of system life-cycles - System development life-cycles, product life-cycles, process models.
Saturday 6th June	Revision
Exam Period	Final Exam

Learning and Teaching Activities

Readings

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

Lectures

Discussion and exploration 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, generate and deliver a system with 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 presentation style is included as part of the student presentation process.

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/

Results

Results shown in *iLearn*, or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in [eStudent](#). For more information visit ask.mq.edu.au.

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 - 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

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

Assessment tasks

- Prac and Research Tasks
- Group Project: Deliverable 1
- Group Project: Deliverable 2
- Project Progress Presentation
- Group Project: Deliverable 3
- Group Project: Deliverable 4
- Final Exam

Learning and teaching activities

- Working in small groups to discuss, generate and deliver a system with documentation for different phases in the software development process.

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

Assessment tasks

- Prac and Research Tasks
- Group Project: Deliverable 1
- Group Project: Deliverable 2
- Project Progress Presentation
- Group Project: Deliverable 3
- Group Project: Deliverable 4
- Final Exam

Learning and teaching activities

- Pre-lecture recommended readings to assist learning concepts and stimulate discussion topics.
- Discussion and exploration of key concepts
- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Group student presentations given by each individual member of the group outlining different aspects investigated in their group-based assignment. Feedback on content and presentation style is included as part of the student presentation process.

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

Assessment tasks

- Prac and Research Tasks
- Group Project: Deliverable 1
- Group Project: Deliverable 2
- Project Progress Presentation
- Group Project: Deliverable 3
- Group Project: Deliverable 4
- Final Exam

Learning and teaching activities

- Discussion and exploration of key concepts
- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Working in small groups to discuss, generate and deliver a system with 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

- Prac and Research Tasks
- Group Project: Deliverable 2
- Group Project: Deliverable 3
- Group Project: Deliverable 4
- Final Exam

Learning and teaching activities

- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Working in small groups to discuss, generate and deliver a system with 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

- Prac and Research Tasks
- Group Project: Deliverable 1
- Group Project: Deliverable 2
- Project Progress Presentation
- Group Project: Deliverable 3
- Group Project: Deliverable 4

- Final Exam

Learning and teaching activities

- Discussion and exploration of key concepts
- Working in small groups to discuss, generate and deliver a system with 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 presentation style is included as part of the student presentation process.

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

Assessment tasks

- Prac and Research Tasks
- Group Project: Deliverable 1
- Group Project: Deliverable 2
- Project Progress Presentation
- Group Project: Deliverable 3
- Group Project: Deliverable 4
- Final Exam

Learning and teaching activities

- Discussion and exploration of key concepts
- Individual tasks and research activities focusing on selected concepts covered in lectures.
- Working in small groups to discuss, generate and deliver a system with documentation for different phases in the software development process.

Changes from Previous Offering

Individual tasks changed from 6 to 4 and the total weighting of individual tasks has been changed from 20% to 30%.

The final exam has been changed from 40% to 30%.

Formal inclusion of requirement (In the grading and standards section) that students must perform at a functional level or better across ALL 3 assessment groupings (Individual tasks, group project deliverables, final exam) in order to pass the unit.

Grading and Standards

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 assessment grouping. 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 raw mark for the course out of 100. Your grade then depends on this total mark and your overall standard of performance.

In order to pass the unit

The Department of Computing expectations are that students have to perform satisfactorily in the final exam as well as in their internal work/assignments. Therefore, in line with this expectation, in order to pass the unit, students must meet the following criteria:

- Perform at a functional level or better across ALL 3 assessment groupings (Individual tasks, group project deliverables, final exam).
- Have satisfactory performance in the final examination (Achieving 40% or lower in the final exam will not be deemed satisfactory).
- Achieve a raw grade of 50 or above.

(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.)

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

- Obtain at least 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 across the assessment groupings.

Four standards (namely: HD, D, CR, P) summarise the different levels of achievement. Each standard is defined to help students know what kind of performance is expected to achieve a certain level of grade. The standards corresponding to the learning outcomes of this unit are given below:

Standards Matrix

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

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

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

Unit guide ITEC800 Systems Engineering Process

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

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

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