

CSC9P6: Software Engineering II**Prerequisites**

CSC9P5: understanding the basic software life-cycle, the role of requirements analysis, specification and design

Credits

22 credits at SCQF level 10

Learning Outcomes

Students should be able to apply techniques for:

- Software Implementation
- Formal Specification

Students should be aware of current ideas and good practice regarding Project Management and Quality Assurance, in the context of software development.

Transferable Skills

- An understanding of the problems of teamwork, and how to work effectively in a team.
- Skills in the design and implementation of systems
- An understanding of the management of engineering complex systems
- The ability to formulate problems in an abstract way

Contents

- Software Implementation 10 Lectures, 4 Practicals
 - Detailed implementation: architecture, associations, state
 - Refactoring designs
 - Creating and exploiting re-usable components
 - Build control, version control, collaborative working
 - Testing, debugging
 - Metrics
- Professional Issues 3 Lectures
 - Topics may include Ethics, Intellectual property rights, Means of protection (patent, copyright, trademark), Legislation relevant to the software professional
- Software Engineering Mathematics and Specification 10 Lectures, 4 Practicals
 - Mathematical Modelling
 - Sets, relations and sequences
 - Logic
 - Formal Specification with Alloy
 - Overview of formal specification languages
- Project Management and Quality Assurance 4 Seminars
 - Topics may Software quality assurance, Process and project metrics, Project Planning, Risk analysis, Risk management, Scheduling and tracking.

Assessment

Project management seminar presentation 5%; Group project (implementation) 25%; Assignment (formal specification) 20%; Examination 50%.

Textbooks

- *Software Engineering* (9th edition), I Sommerville, Pearson, 2010, ISBN 0-137-05346-0 (recommended).
- *Using UML: Software Engineering with Objects and Components* (2nd edition), P Stevens, Addison-Wesley, 2006, ISBN 0-321-26967-5 (recommended).

- *UML Distilled*, 3rd Edition, Martin Fowler, Addison Wesley 2003, ISBN13: 9780321193681, ISBN10: 0321193687
- *Software Abstractions*, 2nd edition, D Jackson, MIT Press, 2011, ISBN 0262017156 (highly recommended)

Requirements

In order to obtain a pass grade for the unit you must

- *Submit all items of assessed work.*
Non-submission of any single item of assessed coursework will result in the award of No Grade for the module as a whole. Assessed coursework submitted late will be accepted up to five days after the submission date (or expiry of any agreed extension) but the grade will be lowered by one grade point per day or part thereof. After five days the piece of work will be deemed a non-submission, and will result in the award of No Grade for the module as a whole. This rule may be relaxed for students who can show good cause for failure to submit. 'Good cause' may include illness (for which a medical certificate or other evidence will be required).
- *Attend the examination.*

If a student is unable to attend the Main examination, he/she must apply to the Student Programmes Office for a Deferred examination. If a Deferred examination is not granted, then the Examiners may allow a Repeat examination.

Students who fail the module but receive a grade of 5C or better will be eligible for a Repeat examination. If a Repeat exam is taken (First Degree Regulations, Regulation 11), then the final grade is obtained from the mark for the Repeat exam together with the original marks for the remaining assessments. The grade obtained following a Repeat exam is capped at 3C, and will not be lower than the original grade.

In addition, Regulation 14 of the University's First Degree Regulations sets out attendance rules for classes that have been defined by the Department as Prescribed. In this module, the Prescribed classes are the tutorials, practicals and group project meetings with a project monitoring officer.