CO2012 Software Project Management and Professionalism

Credits: *10* **Convenor:** *Dr R. Craggs* **Semester:** *1st*

Prerequisites:	*Essential: CO1003, CO1005, CO1007, CO1019*		*Desirable: CO1001, CO1012*		
Assessment:	*Coursework: 100%*				
Lectures:	*9*	*hours*	**Problem Classes:**	*0*	*hours*
Surgeries:	*0*	*hours*	**Class Tests:**	*1*	*hours*
Laboratories:	*13*	*hours*	**Private Study:**	*52*	*hours*

Subject Knowledge

Aims This module will teach you techniques and technologies to run and collaborate on a software project.

Learning Outcomes At the end of the module a student should:

- be able to compare traditional and agile approaches to project planning and monitoring.
- describe the benefits of continuous integration and test automation.
- behave professionally on a software project.
- demonstrate awareness of ethical and legal issues, like the Data Protection Act, likely to affect every professional in the software industry.
- formulate technical problems and their solution in a methodical way;
- research an issue and present their findings in writing in a balanced manner.

Methods Curated pre-lecture videos, lectures, classroom activities, worksheets, supervised labs for mini project group work.

Assessment Marked coursework, including written essay, mini-projects, monitored use of version control and class test.

Skills

Aims To collaborate in a project in a controlled manner working as a team.

Learning Outcomes At the end of the module a student should:

- be able to plan a software project using a traditional approach and an agile approach.
- be able to use git version control;
- be able to apply continuous integration to projects and work producively on projects that use it;
- be able to apply quality control measures to a software project;
- be able to demonstrate what "professionalism" means in the context of the software industry, and be aware of ethical and legal issues, like the Data Protection Act, likely to affect every professional in the software industry.

Methods Class sessions, labs and with worksheets.

Assessment Marked coursework, including a written essay, a mini-project and a class test.

Explanation of Prerequisites A basic knowledge of software development is required. For example the software is created by writing source code which is compiled into software.

Course Description This module teaches the techniques required for working in a team on a software project. It provides practice in applying these within lab exercises and a mini-project done in a group.

Detailed Syllabus

- Project Planning and Gantt Charts
- Team collaboration on software projects;
- Version control using Git
- Continuous integration and automated quality tests
- Agile project planning and monitoring
- The scrum software development framework

Reading List

Pluralsight

”Reading” is provided through curated listed of video content from the Pluralsight library.

Resources Video Content, Slides, study guide, worksheets, lecture rooms with projector.

Module Evaluation Course questionnaires, course review.