Software Engineering (COMP0071)

Description

Aims:
This module covers a range of Software Engineering (SE) material, following on from that introduced in term 1. The emphasis is on the knowledge needed to be able to model, design, implement and evaluate larger software systems effectively. The module starts with development lifecycle models, such as agile development, and then continues to cover requirements specification, requirements analysis, object-oriented concepts, the Unified Modelling Language (UML), and object-oriented design. Software engineering is an inherently practical subject and applying the concepts being taught is a vital component of developing expertise in this area. Consequently students undertake a substantial group project, working through a number of stages of the development of a larger software application. Students will be supervised, but will be expected to largely organize themselves and their work, learning key transferrable skills in team management and organisation.

Learning outcomes:
On successful completion of the module, a student will be able to:
1. Appreciate the wider engineering issues that form the background to developing complex and evolving software-intensive systems;
2. Plan and deliver an effective software engineering process, based on knowledge of widely used development lifecycle models, in particular Agile development;
3. Employ group working skills including general organization, planning and time management and inter-group negotiation;
4. Capture, document and analyse software requirements. Translate a requirements specification into an implementable design, following a structured and organised process;
5. Make effective use of UML, along with design strategies such as defining a software architecture, applying the SOLID principles and using design patterns;
6. Evaluate the quality of the requirements, analysis and

Key information

Year 2019/20
Credit value 15 (150 study hours)
Delivery PGT L7, Campus-based
Reading List View on UCL website
Tutor Dr Ghita Kouadri Mostefaoui
Term Term 2
Timetable View on UCL website

Assessment

For more information about the department, programmes, relevant open days and to browse other modules, visit ucl.ac.uk

Find out more

Disclaimer: All information correct as of June 2019. Please note that aspects of the module may be subject to change. UCL will make best efforts to inform applicants of major changes.
design work done during the module;

**Content:**
Topics covered in lectures during the first half of term:
Course Introduction:
- Overview of the software engineering discipline;
- Software lifecycle models;
- Agile development;
- The Unified Process (UP) Organising development projects; Requirements Engineering:
- Documenting requirements, user stories, use cases and scenarios;
Introduction to UML:
- Review of object-oriented principles;
- UML use case, class, sequence, activity, state, component and deployment diagrams;
- UML models;
The Analysis and Design Process:
- User story realisation;
- Object-oriented modelling;
- Incremental refinement;
Design Principles:
- Software architecture;
- Separation of concerns;
- SOLID Principles;
- Design patterns;
- Object-Oriented design practices;
- Refactoring;
Project Work:
- Project groups are formed during the first week of term, with the early stages of the project running concurrently with the lectures given during the first half of term. The second half of the term is then dedicated to the group project work; Group formation:
- Students are split into project groups with around 4-7 members, each group having a supervisor who acts as the client and also provides support and guidance. Project groups are selected by the module organiser, and each group is allocated a basic project specification to work with. Each project specification gives an outline of an application with a graphical user interface (GUI) to be developed. Project groups are responsible for organising themselves, meeting their supervisor regularly to receive advice and feedback. A group diary should be kept to record the progress of the group, including the minutes of meetings held;
Group deliverables:
- A group report documents the results of the project, focusing on requirements, design, and UML modelling. The group additionally creates a video, lasting around 10 minutes, explaining what their project was about, what were the challenges and giving an overview of the design created. A mock-up of the application user interface is also created to illustrate what the application would look like if fully implemented; Individual deliverables:
- Each student is required to submit an individual report giving a) an assessment of the structure, organisation and progress of the project group, b) an assessment of the project results and the effectiveness of the group as a whole, c) an evaluation of each group member, d) an evaluation of their own role, including a summary of what they contributed.

**Requisites:**
In order to be eligible to select this module, a student must be registered on a programme for which it is a formally-approved option or elective choice.