Software Engineering

Syllabus Information

CS 2800 - Software Engineering

Associated Term: 2022/23 Academic Session
Learning Objectives:

Introducing software Engineering tools and techniques through practical experience of design and development that enable each individual programmer to contribute effective, working, documented code, as part of a team, in a timely fashion Software engineering: models of software development, planning, Object oriented design: Notation for design, identifying objects, classes, attributes and methods. Class relationships, design patterns Programming methodologies: program structure, style, and layout. Coding standards, test driven development Testing: Program analysis, black and white box testing, defensive programming, system, integration and acceptance testing. Use of a variety of modern software engineering tools: in particular version control, debugger, code style checkers, junit and how these can be integrated into an industry standard IDE. Learning Outcomes: understand the software engineering techniques and managerial discipline required to work as part of a team understand and use basic object-oriented concepts appreciate the need for program documentation, testing, readability and modifiability use appropriate tools to support software development: Version control, programming standards, a modern IDE be able to use test driven development to deliver a small scale project
Required Materials: Click here for the reading list system
Technical Requirements: The total number of notional learning hours associated with the course are 150. **These will normally be broken down as follows:** 33 hour(s) of Lecture(s) across 11 week(s) 117 hours of Guided Independent Study **Formative Assessment:** SVN structured commits (300 Minutes) - Graded with Report Moodle Quizzes (200 Minutes) - Moodle feedback **Summative Assessment:** Examination (120 Minutes) - 60% Set Exercise - 10% Set Exercise - 5% Report - 25%