

SWEN30007 Software Systems Project

Credit Points:	12.50											
Level:	3 (Undergraduate)											
Dates & Locations:	This subject is not offered in 2014.											
Time Commitment:	Contact Hours: 36 hours, consisting of one 1-hour lecture and two 1-hour workshops per week Total Time Commitment: 170 hours											
Prerequisites:	<table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>SWEN30006 Software Modelling and Design</td> <td>Semester 1, Semester 2</td> <td>12.50</td> </tr> </tbody> </table>			Subject	Study Period Commencement:	Credit Points:	SWEN30006 Software Modelling and Design	Semester 1, Semester 2	12.50			
Subject	Study Period Commencement:	Credit Points:										
SWEN30006 Software Modelling and Design	Semester 1, Semester 2	12.50										
Corequisites:	None											
Recommended Background Knowledge:	None											
Non Allowed Subjects:	<p>Students cannot enrol in and gain credit for this subject and:</p> <table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>SWEN30004 Software Engineering Project</td> <td>Semester 2</td> <td>12.50</td> </tr> <tr> <td>COMP30016 Computer Science Project</td> <td>Not offered 2014</td> <td>12.50</td> </tr> </tbody> </table> <p>OR</p> <p>433-340 Software Engineering Project</p>			Subject	Study Period Commencement:	Credit Points:	SWEN30004 Software Engineering Project	Semester 2	12.50	COMP30016 Computer Science Project	Not offered 2014	12.50
Subject	Study Period Commencement:	Credit Points:										
SWEN30004 Software Engineering Project	Semester 2	12.50										
COMP30016 Computer Science Project	Not offered 2014	12.50										
Core Participation Requirements:	<p><p>For the purposes of considering request for Reasonable Adjustments under the Disability Standards for Education (Cwth 2005), and Student Support and Engagement Policy, academic requirements for this subject are articulated in the Subject Overview, Learning Outcomes, Assessment and Generic Skills sections of this entry.</p> <p>It is University policy to take all reasonable steps to minimise the impact of disability upon academic study, and reasonable adjustments will be made to enhance a student's participation in the University's programs. Students who feel their disability may impact on meeting the requirements of this subject are encouraged to discuss this matter with a Faculty Student Adviser and Student Equity and Disability Support: http://services.unimelb.edu.au/disability</p></p>											
Contact:	email: tmiller@unimelb.edu.au (mailto:tmiller@unimelb.edu.au)											
Subject Overview:	<p>AIMS</p> <p>The aim of the subject is to give students an understanding of how and when to apply their knowledge of computer science and software engineering in the development of a non-trivial software system. The subject also aims to give students an understanding of the importance of analysis, design and management activities within a development process and to provide a realistic environment in which students understand how the practical aspects of computer science and software engineering are applied to real world projects.</p> <p>INDICATIVE CONTENT</p> <p>Students will work in teams to analyse, design, implement and test a non-trivial software system for a realistic client. A key part of the project is for students to develop and manage a repeatable process in order to deliver a quality software product Workshops will explore the application of theory to the project and include selected topics drawn from: requirements analysis, design, implementation, testing and software project management relevant to the phase of the project that students are currently working on.</p>											

Learning Outcomes:	<p>INTENDED LEARNING OUTCOMES (ILO)</p> <p>On completion of this subject the student is expected to:</p> <ol style="list-style-type: none"> 1 Be able to analyse, design implement and test a non-trivial software system 2 Be able to undertake problem identification, formulation and solution 3 Be able to communicate effectively, not only with engineers but also with the community at large 4 Be able to apply software engineering principles to the development of non-trivial projects
Assessment:	<p>The subject is assessed on the project management, problem analysis, software design, implementation and testing, artefacts generated during the project and submitted at the end of the project, and on a final report submitted by the team at the end of the project. 1. The analysis and process-related documentation totalling approximately 8000-10000 words (55%) and includes: A project management component assessing the team's ability to manage a repeatable process, reviewed in week 3 An analysis and design component assessing the team's ability to apply software engineering principles and techniques to the development of a software system, reviewed in weeks 5 and 7 and 9 A test planning component assessing the team's ability to apply software engineering knowledge of validation and verification, reviewed in weeks 7 and 9. 2. A final release, of Software, submitted in week 12, requiring approximately 45 hours of work (30%) that assesses the team's ability to develop a non-trivial software system using software engineering principles and techniques 3. A component of the marks will be based on the individual's contribution to the project over the course of the semester (15%) A component of each submission addresses each of the Intended Learning Outcomes (ILOs) ILOs 1, 2, and 4 are specifically addressed by the final release (item 2) ILO 3 is specifically addressed via the project management component of item 1</p>
Prescribed Texts:	None
Breadth Options:	This subject is not available as a breadth subject.
Fees Information:	Subject EFTSL, Level, Discipline & Census Date, http://enrolment.unimelb.edu.au/fees
Generic Skills:	<p>On completion of this subject students should have developed the following skills:</p> <ul style="list-style-type: none"> # Ability to apply knowledge of science and software engineering fundamentals # Ability to undertake problem identification, formulation, and solution # Proficiency in software engineering design # Ability to conduct an engineering project # Ability to communicate effectively, with the engineering team and with the community at large # Ability to manage information and documentation # Capacity for creativity and innovation # Understanding of professional and ethical responsibilities, and commitment to them # Ability to function effectively as an individual and in multidisciplinary and multicultural teams, as a team leader or manager as well as an effective team member # Capacity for lifelong learning and professional development
Notes:	<p>LEARNING AND TEACHING METHODS</p> <p>The software systems project provides a realistic learning environment with a realistic product specification.</p> <p>Tutors in the subject act as mentors and guide teams throughout the project. The interaction between the student team and the tutors often raise issues that provide the topics for workshops.</p> <p>The subject comprises one lecture and one two-hour workshop per week. Lectures are used to coordinate the teams, deliver software engineering theory and practice relevant to the stage of the project reached, and to share experiences between the teams. Workshops are used to discuss issues raised within the project, translate theory to practice relevant to the stage of the project reached, to provide hands-on practice with tools, and to share experiences.</p>

INDICATIVE KEY LEARNING RESOURCES

The subject is administered through the Universities Learning Management System. Templates for the various artefacts, guidelines on engineering processes and links to software engineering tools are available through the LMS. A standard development environment is provided that includes programming languages, libraries and development tools is provided to the students and is available on most engineering computers.

CAREERS / INDUSTRY LINKS

The software industry is expanding and along with it the demand for software engineers that are capable of the analytical and management skills beyond programming. The industry is also changing in the nature of the projects being undertaken with many software engineers now working in multidisciplinary project teams. The skills and experience gained in this subject are valued by employers and are often seen as a necessary grounding for a career in software and technology related industries.

The subject aims to source product ideas from clients outside of the Department where possible and thus seeks to expose students to the types of environments in which software engineering projects take place. Guest lectures by are also given to highlight aspects of industrial practice and to expose students to the practical aspects of software engineering.

**Related Majors/Minors/
Specialisations:**

Science-credited subjects - new generation B-SCI and B-ENG.
Selective subjects for B-BMED