

SWEN90013 Masters Advanced Software Project

Credit Points:	25											
Level:	9 (Graduate/Postgraduate)											
Dates & Locations:	2015, Parkville This subject commences in the following study period/s: Year Long, Parkville - Taught on campus.											
Time Commitment:	Contact Hours: 48 hours, comprising of one 2-hour workshop over 24 weeks. Total Time Commitment: 400 hours											
Prerequisites:	<table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>SWEN90014 Masters Software Engineering Project</td> <td>Semester 2</td> <td>12.50</td> </tr> <tr> <td>SWEN90006 Software Testing and Reliability</td> <td>Semester 2</td> <td>12.50</td> </tr> </tbody> </table>			Subject	Study Period Commencement:	Credit Points:	SWEN90014 Masters Software Engineering Project	Semester 2	12.50	SWEN90006 Software Testing and Reliability	Semester 2	12.50
Subject	Study Period Commencement:	Credit Points:										
SWEN90014 Masters Software Engineering Project	Semester 2	12.50										
SWEN90006 Software Testing and Reliability	Semester 2	12.50										
Corequisites:	None											
Recommended Background Knowledge:	Completion of the following subject prior to enrolling in this subject is recommended:											
	<table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>SWEN40004 Modelling Complex Software Systems</td> <td>Semester 1</td> <td>12.50</td> </tr> </tbody> </table>			Subject	Study Period Commencement:	Credit Points:	SWEN40004 Modelling Complex Software Systems	Semester 1	12.50			
Subject	Study Period Commencement:	Credit Points:										
SWEN40004 Modelling Complex Software Systems	Semester 1	12.50										
Non Allowed Subjects:	Students cannot enrol in and gain credit for this subject and:											
	<table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>SWEN40001 Advanced Software Engineering Project</td> <td>Year Long</td> <td>25</td> </tr> </tbody> </table>			Subject	Study Period Commencement:	Credit Points:	SWEN40001 Advanced Software Engineering Project	Year Long	25			
Subject	Study Period Commencement:	Credit Points:										
SWEN40001 Advanced Software Engineering Project	Year Long	25										
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>											
Coordinator:	Dr Edmund Kazmierczak											
Contact:	email: edmundak@unimelb.edu.au (mailto:edmundak@unimelb.edu.au)											
Subject Overview:	<p>AIMS</p> <p>The aim of the subject is to give the students the knowledge and skills required to carry out real life software engineering projects. Students will work in large teams to develop a non-trivial software system for an external client using advanced software engineering techniques. In this subject managing the quality built into the final product is extremely important and students will be expected develop and manage processes that achieve high levels of quality and dependability. Workshops are used to explore the application of advanced software engineering techniques to student projects and are drawn from topics in: analysis and modeling, product and project metrics, design and technologies, product testing and measurement and validation.</p>											

	<p>INDICATIVE CONTENT</p> <p>Developing real-world software on time and under budget is a challenging task. Students will work in a team to solve a practical problem, applying sound engineering principles to the formulation and solution of their problem. Students will engage in the full software engineering life cycle from requirements engineering through to delivery, to develop a software solution for an external client.</p> <p>This subject has been integrated with the Skills Towards Employment Program (STEP) and contains activities that can assist in the completion of the Engineering Practice Hurdle (EPH).</p>
<p>Learning Outcomes:</p>	<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 design analyse, implement, test and deliver real-world software systems 2 Be able to design software engineering processes appropriate to a specific problem or set of non-functional requirements 3 Be able to research, understand and apply cutting edge computing technology to the solution of complex problems in software engineering 4 Be able to work with external stakeholders to develop quality requirements specifications 5 Be able to effectively manage large teams utilising a variety of software engineering processes
<p>Assessment:</p>	<p>Each project team of 10 or more students is required to submit artefacts detailing their project management, analysis and design. Each team will be asked to make a number of presentations during the year that are not assessed but will be used to give feedback. Assessment Components Each individual student's mark will be made up of the following components: The quality of the engineering work as demonstrated by an individual student in the work that they produce as required by the project plan and the deliverables submitted at the end the project, requiring approximately 120 - 175 hours of work (40%). Depending on the project plan and particular roles a student performs, this could involve work such as requirements analysis, product design, software design, programming, software verification, and product evaluation The contribution of an individual student to the planning, management and governance of the project and their ability to function as an effective member of a team, requiring approximately 120 - 175 hours of work (40%). Depending on the project plan and particular roles a student performs, this could involve work such as project scheduling, risk management, configuration management, quality assurance, team presentations, client liaison, meeting participation, and team leadership A team mark for the quality of the final delivered product and supporting documentation (20%). Key Submissions for the Project A systematic and disciplined approach to the development of software requires the selection of good processes, the ability to work in teams and the production of a number of documents and other artefacts as part of the engineering process. The engineering artefacts produced as part of the project as well as the final set of deliverables is used as the basis for assessment. Specifically: A project management component outlining the team's processes, to be submitted for review in week 4 of semester 1. Requirements analysis, software design, code and testing components assessing the team's ability to apply software engineering principles and techniques to the disciplined and systematic development of a software system, submitted with respect to the timeline identified in item 1 A final release of Software, submitted in week 10 that assesses the team's ability to develop a non-trivial software system using software engineering principles and techniques. All students are expected to participate in team meetings and workshops and contribute towards analysis and process related documentation, coding and other team activities. Each student is expected to spend around 350 hours during the year for project activities outside of workshops with subject staff. The split between engineering work and project management is dependent on the roles performed by the student. Intended Learning Outcomes (ILOs) 1, 2, 3, 4 and 5 are assessed by the student's contributions to the engineering artefacts produced in item 2. To pass the subject, students must obtain at least 50% overall.</p>
<p>Prescribed Texts:</p>	<p>TBA</p>
<p>Breadth Options:</p>	<p>This subject is not available as a breadth subject.</p>
<p>Fees Information:</p>	<p>Subject EFTSL, Level, Discipline & Census Date, http://enrolment.unimelb.edu.au/fees</p>

Generic Skills:	<p>On completion of this subject students should have the following skills:</p> <ul style="list-style-type: none"> # Ability to apply knowledge of science and engineering fundamentals # Ability to undertake problem identification, formulation, and solution # Ability to utilise a systems approach to complex problems and to design and operational performance # 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.
Notes:	<p>LEARNING AND TEACHING METHODS</p> <p>The advanced software engineering project provides students with a realistic learning environment typically with an external client that wants to use the software. Teams are large and often consist of ten or more students.</p> <p>Each student team is assigned a staff supervisor who acts as a mentor and guide throughout the project. Teams meet with their supervisor regularly to monitor progress and to discuss issues that arise during the project.</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 often in the form of short presentations. Workshops are used to discuss issues relevant to the team, and to help the team plan and manage their engineering processes.</p> <p>INDICATIVE KEY LEARNING RESOURCES</p> <p>The subject is administered through the Universities Learning Management System. Papers, tools and links to engineering processes and 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.</p> <p>CAREERS / INDUSTRY LINKS</p> <p>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 highly valued by employers and are often seen as a vital and necessary grounding for a career in software and technology related industries.</p> <p>The subject aims to source projects from clients outside of the University 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.</p>
Related Majors/Minors/ Specialisations:	<p>Master of Engineering (Software with Business) Master of Engineering (Software)</p>