

SWEN30006 Software Modelling and Design

SWEN20003 Object Oriented Software Development and Design

Credit Points:	12.5																					
Level:	3 (Undergraduate)																					
Dates & Locations:	2015, Parkville This subject commences in the following study period/s: Semester 1, Parkville - Taught on campus. Semester 2, Parkville - Taught on campus.																					
Time Commitment:	Contact Hours: 48 hours, comprising of two 1-hour lectures and one 2-hour workshop per week Total Time Commitment: 170 hours																					
Prerequisites:	<p>One subject from Group A and one subject from Group B.</p> <p>Group A</p> <table><tr><th>Subject</th><th>Study Period Commencement:</th><th>Credit Points:</th></tr><tr><td>SWEN20003 Object Oriented Software Development</td><td>Semester 2</td><td>12.50</td></tr><tr><td>COMP90041 Programming and Software Development</td><td>Semester 1, Semester 2</td><td>12.50</td></tr></table> <p>OR</p> <p>433-254 Software Design</p> <p>Group B</p> <table><tr><th>Subject</th><th>Study Period Commencement:</th><th>Credit Points:</th></tr><tr><td>COMP20003 Algorithms and Data Structures</td><td>Semester 2</td><td>12.50</td></tr><tr><td>COMP90038 Algorithms and Complexity</td><td>Semester 1, Semester 2</td><td>12.50</td></tr><tr><td>COMP20007 Design of Algorithms</td><td>Semester 1</td><td>12.50</td></tr></table> <p>OR</p> <p>433-253 Algorithms and Data Structures</p>	Subject	Study Period Commencement:	Credit Points:	SWEN20003 Object Oriented Software Development	Semester 2	12.50	COMP90041 Programming and Software Development	Semester 1, Semester 2	12.50	Subject	Study Period Commencement:	Credit Points:	COMP20003 Algorithms and Data Structures	Semester 2	12.50	COMP90038 Algorithms and Complexity	Semester 1, Semester 2	12.50	COMP20007 Design of Algorithms	Semester 1	12.50
Subject	Study Period Commencement:	Credit Points:																				
SWEN20003 Object Oriented Software Development	Semester 2	12.50																				
COMP90041 Programming and Software Development	Semester 1, Semester 2	12.50																				
Subject	Study Period Commencement:	Credit Points:																				
COMP20003 Algorithms and Data Structures	Semester 2	12.50																				
COMP90038 Algorithms and Complexity	Semester 1, Semester 2	12.50																				
COMP20007 Design of Algorithms	Semester 1	12.50																				
Corequisites:	None																					
Recommended Background Knowledge:	None																					
Non Allowed Subjects:	433-341 Software Engineering Process & Practice																					
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: karus@unimelb.edu.au (mailto:karus@unimelb.edu.au)
Subject Overview:	<p>AIMS</p> <p>Software Systems must be carefully analysed, designed and modelled before they are constructed. This subject delivers the knowledge and skills needed for the design of medium to large-scale software systems. The subject also teaches students how off-the-shelf development frameworks can be utilized for designing large-scale software systems. The emphasis will be on techniques appropriate for object-oriented design and development.</p> <p>INDICATIVE CONTENT</p> <p>Topics covered include:</p> <ul style="list-style-type: none"> # Analysis of requirements for design # Modelling software systems using the Unified Modelling Language (UML) # Software design processes and principles # Common design patterns and software architectures # Tools for design and development.
Learning Outcomes:	<p>INTENDED LEARNING OUTCOMES (ILO)</p> <p>On completion of this subject the student is expected to be able to:</p> <ol style="list-style-type: none"> 1 Analyse requirements 2 Carry out an architectural and detailed design for medium-sized software systems 3 Select appropriate design patterns for a use in a design 4 Use a given implementation platform and framework to develop a software system
Assessment:	<p>A single team based project will be undertaken in teams of 3 during the semester. The project work is expected to take approximately 35 – 40 hours of work (40%) per student. The project requires the analysis, design and implementation of a small application. The project will be assessed in three stages and due at the end of week 4, requiring approximately 15 - 20 hours of work per student and due at the start of week 8, and due at the end of week 11. ILO's 1, 2, 3 and 4 are addressed in the project. The project is a hurdle and must be passed to pass the subject. One two hour written closed book end-of-semester written examination (60%). ILO's 1, 2, 3, and 4 are assessed in the examination. The examination is a hurdle and must be passed, to pass the subject. Hurdle requirement: To pass the subject, students must obtain at least 50% overall and obtain a pass in the project work and a pass in the examination. Intended Learning Outcomes (ILOs) 1, 2 and 3 are addressed in the lectures, workshop exercises, project assignments and the end-of-semester examination. ILO 4 is addressed in the workshop exercises and project assignment semester test is not a hurdle and students are not required to pass the mid-semester test to pass the subject.</p>
Prescribed Texts:	None
Breadth Options:	<p>This subject potentially can be taken as a breadth subject component for the following courses:</p> <ul style="list-style-type: none"> # Bachelor of Arts (https://handbook.unimelb.edu.au/view/2015/B-ARTS) # Bachelor of Commerce (https://handbook.unimelb.edu.au/view/2015/B-COM) # Bachelor of Environments (https://handbook.unimelb.edu.au/view/2015/B-ENVS) # Bachelor of Music (https://handbook.unimelb.edu.au/view/2015/B-MUS) <p>You should visit learn more about breadth subjects (http://breadth.unimelb.edu.au/breadth/info/index.html) and read the breadth requirements for your degree, and should discuss your choice with your student adviser, before deciding on your subjects.</p>
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 generic skills</p> <ul style="list-style-type: none"> # Ability to undertake problem identification, formulation and solution # Proficiency in engineering design

	# Ability to utilise a systems approach to design and operational performance.
Notes:	<p>LEARNING AND TEACHING METHODS</p> <p>The subject comprises two one lectures and one two hour workshop each week. Weekly readings are assigned from the textbook. The subject also includes a design/implementation project, which involves analysis, design, implementation and delivery.</p> <p>INDICATIVE KEY LEARNING RESOURCES</p> <p>At the beginning of the year, the coordinator will propose a textbook that will be made available through University Book Shop and library. The current suggested textbook is Christopher Fox: Introduction to Software Engineering Design, Processes, Principles, and Patterns with UML2. Additional learning material will be made available on the learning management system (LMS) site for the subject.</p> <p>CAREERS / INDUSTRY LINKS</p> <p>The software industry is a large and steadily growing industry, and is constantly looking for competent software engineers. This subject teaches the software engineering design principles and core software design skills required by industry practitioners.</p>
Related Course(s):	Diploma in Informatics Master of Information Technology
Related Majors/Minors/ Specialisations:	B-ENG Software Engineering stream Computer Science Computer Science Computer Science Computer Science Computer Science Computing and Software Systems MIT Computing Specialisation Master of Engineering (Mechatronics) Master of Engineering (Software with Business) Master of Engineering (Software) Science-credited subjects - new generation B-SCI and B-ENG. Selective subjects for B-BMED
Related Breadth Track(s):	Computing