

## 433-342 Software Engineering Methods

<b>Credit Points:</b>	12.50
<b>Level:</b>	3 (Undergraduate)
<b>Dates &amp; Locations:</b>	2009, This subject commences in the following study period/s: Semester 2, - Taught on campus.
<b>Time Commitment:</b>	Contact Hours: Twenty-four hours of lectures and approximately 11 hours of tutorials Total Time Commitment: Not available
<b>Prerequisites:</b>	433-341 Software Engineering Process and Practice
<b>Corequisites:</b>	None
<b>Recommended Background Knowledge:</b>	None
<b>Non Allowed Subjects:</b>	None
<b>Core Participation Requirements:</b>	<p>&lt;p&gt;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.&lt;/p&gt;         &lt;p&gt;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: &lt;a href="http://services.unimelb.edu.au/disability"&gt;http://services.unimelb.edu.au/disability&lt;/a&gt;&lt;/p&gt;</p>
<b>Coordinator:</b>	Dr Timothy Miller
<b>Subject Overview:</b>	Software Engineering Methods aims to examine some of the more specific processes required for the production of high quality software, for example, methods for testing software, ensuring reliability or performance in software.
<b>Objectives:</b>	<p>On completion of this subject students should:</p> <ul style="list-style-type: none"> <li># understand the principles of software testing;</li> <li># know how to apply software testing techniques to the development of quality software;</li> <li># understand the principles of software reliability and methods for assessing software reliability; and</li> <li># understand and apply a range of engineering methods.</li> </ul>
<b>Assessment:</b>	Two written assignments of approximately 5 pages per assignment including figures and tables (10% each); a project and project report, of approximately 20 pages (30%); and a 2-hour end-of-semester written examination (50%). To pass the subject, students must obtain at least 50% overall, 10/20 in assignments, 15/30 in project work, and 25/50 in the written examination.
<b>Prescribed Texts:</b>	None
<b>Recommended Texts:</b>	Information Not Available
<b>Breadth Options:</b>	This subject is not available as a breadth subject.
<b>Fees Information:</b>	Subject EFTSL, Level, Discipline & Census Date, <a href="http://enrolment.unimelb.edu.au/fees">http://enrolment.unimelb.edu.au/fees</a>
<b>Generic Skills:</b>	<p>On completion of this subject students should have:</p> <ul style="list-style-type: none"> <li># the ability to apply knowledge of basic science and engineering fundamentals;</li> </ul>

	<ul style="list-style-type: none"><li># in-depth technical competence in at least one engineering discipline;</li><li># the ability to undertake problem identification, formulation and solution; and</li><li># the ability to utilise a systems approach to design and operational performance.</li></ul>
<b>Related Course(s):</b>	Bachelor of Engineering (Computer Engineering) Bachelor of Engineering (Electrical Engineering) Bachelor of Engineering (EngineeringManagement) Software Bachelor of Engineering (Mechatronics) and Bachelor of Computer Science Bachelor of Engineering (Software Engineering)
<b>Related Majors/Minors/ Specialisations:</b>	Computer Science Major