Subject Description Form

<table>
<tr><td>Subject Code</td><td>COMP5212</td></tr>
<tr><td>Subject Title</td><td>Software Design and Architecture</td></tr>
<tr><td>Credit Value</td><td>3</td></tr>
<tr><td>Level</td><td>5</td></tr>
<tr><td>Pre-requisite/ Exclusion</td><td>Nil</td></tr>
<tr><td>Objectives</td><td>The objectives of this subject are to:

1. introduce design concepts such as abstraction, information hiding, functional decomposition, modularization and reusability;
2. provide an opportunity for students to learn how to cope with the complexity of problem specification, make design trade-offs, and use software architecture and domain knowledge for development.</td></tr>
<tr><td>Intended Learning Outcomes</td><td>Upon completion of the subject, students will be able to:

a) apply abstraction, information hiding, functional decomposition and modularization in design;
b) define a software architecture;
c) understand design tradeoffs and apply various design representations;
d) use basic design methods; and
e) use design metrics to evaluate a design.</td></tr>
<tr><td>Subject Synopsis/ Indicative Syllabus</td><td>• Concepts and Principles
• Design Notations
• Design Methods: Object-Oriented Design, Aspect-Oriented Design and Programming, Component-based Development, Commercial Off-the-Shelf, Rational Unified Software, UML
• Design Quality and Metrics
• Software Architecture
• Design Strategies and Method
• Psychology of Programming</td></tr>
<tr><td>Teaching/Learning Methodology</td><td>Lectures, self study, face-to-face/online tutorials, discussion forums, lab/workshop/seminar where applicable</td></tr>
<tr><td>Assessment Methods in Alignment with Intended Learning Outcomes</td><td>
<table>
<tr><td rowspan="2">Specific Assessment Methods/Tasks</td><td rowspan="2">% weighting</td><td colspan="5">Intended subject learning outcomes to be assessed</td></tr>
<tr><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td></tr>
<tr><td>Assignments, Tests & Projects</td><td>55</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr>
<tr><td>Final Examination</td><td>45</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr>
<tr><td>Total</td><td>100</td><td colspan="5"></td></tr>
</table>
</td></tr>
</table>

Student study effort expected	**Class Contact:**	
	Class activities (lecture, tutorial, lab)	39 hours
	Other student study effort:	
	Assignments, Quizzes, Projects, Exams	65 hours
	Total student study effort	**104 hours**
Reading list and references	(1) Bass, L., Clements, P., Kazman, R., 2013, Software architecture in practice , 3rd Ed, Addison-Wesley. (2) Rozanski, N., Woods, E., 2011, Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives, 2nd Ed, Addison-Wesley. (3) Detienne, F., Bott, F., 2013, Software design: cognitive aspects, Springer. (4) Bennett, S., Ray, F., 2010, Object-Oriented Systems Analysis and Design Using UML, 4th Ed, McGraw-Hill.	