

Subject Description Form

Subject Code	COMP306
Subject Title	Software Engineering and User Interface
Credit Value	3
Level	3
Pre-requisite / Co-requisite/ Exclusion	Nil
Objectives	<ul style="list-style-type: none"> • To introduce students a general knowledge of the application of software engineering techniques in different stages and aspects of software development. • To provide students with knowledge and understanding of interface architecture, and design, development and evaluation of user interface.
Intended Learning Outcomes	<p>Upon completion of the subject, students will be able to:</p> <p><i>Professional/academic knowledge and skills</i></p> <p>(a) apply software engineering techniques in the systems specifications and design stages of software projects;</p> <p>(b) apply software engineering techniques to real-life projects;</p> <p>(c) understand the user's requirements in human computer interface design and be able to collect such requirements;</p> <p>(d) design a professional human computer interface according to the user's requirements using suitable technologies;</p> <p>(e) evaluate a human computer interface design;</p> <p><i>Attributes for all-roundedness</i></p> <p>(f) solve problems by using systematic approaches;</p> <p>(g) communicate effectively through report writing and project presentations.</p> <p>Alignment of Programme Outcomes:</p> <p>Program Outcome 1. Practice communication skill in discussion and project presentation and documentation ;</p> <p>Program Outcome 3. Understand and value ethical issues in user interface design and development of computing systems;</p> <p>Program Outcome 4. Think and reason critically on developing alternatives in problem solving and user interface development, and be able to design and</p>

	<p>evaluate the proper user interface by applying related technologies;</p> <p>Program Outcome 5. Possess technical knowledge needed to solve design and user interface problems and to realize solutions in programming user interface technology;</p> <p>Program Outcome 7. Work together as a team in project design and development;</p> <p>Program Outcome 8. Solve problems and develop solutions with design and user interface technologies in different application areas.</p>
--	---

Subject Synopsis/ Indicative Syllabus	Topic	Duration of Lectures
	1. Software specification Requirements specification; formal specification; algebraic specification; model-based specification.	3
	2. Software design Object-oriented design; function-oriented design; real-time systems design.	4
	3. Software management Project planning and scheduling; software cost estimation; documentation; software quality assurance.	4
	4. Dialogue interactions and structures Dialogue interaction: types and techniques, navigation and orientation, multimedia and non-graphical dialogues; dialogue issues: response time, control, standards, look and feel; use of metaphors for describing interface behaviour; statistical models for describing interaction processes; computer graphics, color representation; layers model of architecture of design and windowing systems; windows manager model (e.g. X, Macintosh and Microsoft windows).	6
	5. User interface design Graphic design basics: design languages, typography, use of color, 2D and 3D spatial organization, temporal sequencing; user-centred design and task analysis; participatory design and prototyping; non-formal cognitive approach.	5
	6. User interface development Dialogue toolkits (e.g. MacApp, NextStep, HyperCards, windows and Visual Basic); user interface management system.	3
	7. User interface evaluation Evaluation techniques: productivity and figures of time, errors, tameability, preference; usability testing techniques: field observation, psychometric methods, video, protocols, experiment design; example systems and case studies: MS DOS, Apple Macintosh, MacPaint, HyperCard and Multimedia systems, UNIX, dBASE IV.	3

Total	28

Laboratory Experiment: Use of HCI tools

Teaching/Learning Methodology

Lectures focus on introduction and explanation of key concepts and techniques. Tutorial and lab sessions provide students opportunity to practice the techniques and tools presented in class. Assignments and project allow students to deepen their understanding of the concepts taught in class and apply the theory and techniques to software design and user interface development. Students will be encouraged to work in groups to share and present ideas, review other’s work, and develop teamwork skill.

Assessment Methods in Alignment with Intended Learning Outcomes

Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed (Please tick as appropriate)						
		a	b	c	d	e	f	g
1. Assignments	55%	✓	✓	✓	✓	✓	✓	
2. Lab exercises		✓	✓		✓	✓	✓	✓
3. Project		✓	✓		✓		✓	✓
4. Mid-term		✓	✓		✓	✓	✓	
5. Examination	45%	✓	✓		✓	✓	✓	
Total	100 %							

Explanation of the appropriateness of the assessment methods in assessing the intended learning outcomes:

Assignments, project and mid-term test act as a measure on the understandings of the students on the basic concepts of the software specification, design and user interface design and evaluation.

In addition, project will be used to measure the understandings of the students about the current practice in software design and user interface design. The students can improve their presentation and communication skills through the project presentation, and practice team work. Students can also develop their analytic and problem solving skills.

Examination will be used as an overall measure of the understandings of the students on specification and design process, software design and user interface concepts and technologies.

Student Study Effort Required

Class contact:	
▪ Lecture	28 Hrs.
▪ Tutorial/Lab	14 Hrs.
Other student study effort:	

	<ul style="list-style-type: none"> ▪ Work on assignments/ reading related material/ group discussion 	65 Hrs.
	<ul style="list-style-type: none"> ▪ Study for mid-term and examination 	25 Hrs.
	Total student study effort	132 Hrs.
Reading List and References	Reference Books: <ol style="list-style-type: none"> 1. Sommerville, I., Software Engineering, 9th Edition, Addison-Wesley, 2010. 2. Ghezzi, C. Jazayeri, M. and Mandrioli, D., Fundamentals of Software Engineering, Second Edition, Prentice Hall, 2003. 3. Dix, A., Finlay, J., Abowd, G. and Beale, R., Human-Computer Interaction, Third Edition, Prentice Hall, 2004. 	