

Course Syllabus

COMP S350F Software Engineering

Effective Period: 2021 Autumn -

1. Course Data

Credits	: 5
Medium of Instruction	: English
Contact Hours	: 52
Course Level	: Higher
Pre-requisite	: Two 5-credit computer programming courses
Excluded combination	: None

2. Introduction

COMPS350F is a 5-credit higher level course offered by the School of S&T in face-to-face mode. It is a core course in Bachelor of Computing with Honours in Internet Technology, Bachelor of Science with Honours in Computer Science, Bachelor of Arts with Honours in Computing & Interactive Entertainment and Bachelor of Science with Honours in Computer Engineering. It is also an elective course in Bachelor of Engineering with Honours in Electronic and Computer Engineering.

3. Aims and Course Intended Learning Outcomes (CILOs)

Upon completion of this course, students should be able to:

1. Explain software engineering concepts and activities.
2. Create software development project artifacts (e.g. software requirements specification / SRS).
3. Design software systems to meet user requirements.
4. Select a suitable software development process based on requirements and resources.
5. Apply quality assurance methods (e.g. testing and inspection).
6. Make ethical decisions playing the role of a software engineer.

4. Course Contents

Tentative schedule

Week	Class Activity	CILOs
1	Syllabus, and introduction to SE	1-6
2	System engineering	1-6
3	Domain modeling (UML)	1-3,6
4	Architectural design	1-3,6
5	Use case modeling	1-3
6	Object-interaction modeling	1-4
7	User interface design	1-3,6
8	Deriving a design class diagram	1-4,6
9	Software testing	4-6
10	Software quality assurance	1-6
11	Software maintenance	4-6
12	Software implementation	4-6
13	Software project management	1-6
14	Team project presentation (TBC)	1-6

5. Assessment

<i>Assessment</i>		<i>Weighting</i>
OCAS (individual work)	Individual activities / documents / tests	30%
	Reflective learning journal	20%
OES (group work)	Group documents/prototypes	50%

6. Textbook and References

6.1 Textbooks

- David Kung, “Object-Oriented Software Engineering: An Agile Unified Methodology”, McGraw-Hill Higher Education, 2013.

6.2 References

- Rod Stephens, “Beginning Software Engineering”, Wrox, 2015.
- Gojko Adzic, “Specification by Example”, Manning Publications, 2011.
- Pierre Bourque and Richard E. Fairley (Eds.), “SWEBOK V3.0 – Guide to the Software Engineering Body of Knowledge”, IEEE Computer Society, 2014.