

CS5351: SOFTWARE ENGINEERING

Effective Term

Semester A 2025/26

Part I Course Overview

Course Title

Software Engineering

Subject Code

CS - Computer Science

Course Number

5351

Academic Unit

Computer Science (CS)

College/School

College of Computing (CC)

Course Duration

One Semester

Credit Units

3

Level

P5, P6 - Postgraduate Degree

Medium of Instruction

English

Medium of Assessment

English

Prerequisites

CS2312 Problem Solving and Programming or equivalent

Precursors

Nil

Equivalent Courses

Nil

Exclusive Courses

Nil

Part II Course Details

Abstract

The aim of this graduate-level course is to provide students with a comprehensive understanding of the state-of-art and practices in the software engineering (SE) discipline, its associated processes/methodologies and current trends.

This includes in-depth coverage of some of the key SE issues, best practices and guidelines and an overview of project management techniques. The key objective is to equip students with SE knowledge so that they will be able to take full advantage of these concepts, processes, and best practices in their future software development projects.

Course Intended Learning Outcomes (CILOs)

CILOs		Weighting (if DEC-A1 app.)		DEC-A2	DEC-A3
1	Describe fundamental software engineering process models and comprehend the current trends		x		
2	Explain and analyse advanced software engineering principles and techniques			x	
3	Create and maintain the design and codebase of software engineering projects			x	
4	Perform team-based software engineering tasks.		x		

A1: Attitude

Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability

Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

A3: Accomplishments

Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

Learning and Teaching Activities (LTAs)

LTAs		Brief Description	CILO No.	Hours/week (if applicable)
1	Lectures	Students will engage with selected software engineering methodologies and advanced techniques and principles. These elements will be illustrated with principles, examples, and demonstration.	1, 2, 3	2 hours/ week
2	Tutorials	Students will discuss and practice various principles and skills in software developments and project management in a controlled software engineering context.	1, 2, 3	1 hour/ week

3	Reading	Students will read from assigned software engineering materials and explain the technicality with original ideas for advancing the state of the art.	1, 2	
4	Group Project	Students as a team will design and create a system for software engineering projects. They will collaborate and share in their learning process. They will also practice materials in all major topics from project management and requirements to coding and testing in the project.	1, 2, 3, 4	

Assessment Tasks / Activities (ATs)

ATs		CILO No.	Weighting (%)	Remarks ("- for nil entry)	Allow Use of GenAI?
1	Quiz	1, 2, 3	20	-	No
2	Research Paper Presentation	2	10	-	No
3	Project	1, 2, 3, 4	30	-	Yes

Continuous Assessment (%)

60

Examination (%)

40

Examination Duration (Hours)

2

Minimum Examination Passing Requirement (%)

30

Additional Information for ATs

For a student to pass the course, at least 30% of the maximum mark for the examination must be obtained.

Assessment Rubrics (AR)**Assessment Task**

Quiz (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion

1.1 ABILITY to describe, analyse and apply software engineering processes and techniques.

Excellent

(A+, A, A-) High

Good

(B+, B, B-) Significant

Fair

(C+, C, C-) Moderate

Marginal

(D) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Research Paper Presentation (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion

2.1 ABILITY to explain and present technical contents in software engineering research papers.

Excellent

(A+, A, A-) High

Good

(B+, B, B-) Significant

Fair

(C+, C, C-) Moderate

Marginal

(D) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Project (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion

3.1 ABILITY to apply software engineering techniques and good practices for actual software development.

3.2 ABILITY to use project management skills in team project.

3.3 ABILITY to report in an organised and logical way. All works are professional presented. All sources are correctly and thoroughly documented.

Excellent

(A+, A, A-) High

Good

(B+, B, B-) Significant

Fair

(C+, C, C-) Moderate

Marginal

(D) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Examination (for students admitted before Semester A 2022/23 and in Semester A 2024/25 & thereafter)

Criterion

- 4.1 ABILITY to explain software development processes and compare process models.
- 4.2 ABILITY to apply software analysis and design techniques.
- 4.3 ABILITY to apply project management techniques on given case study.
- 4.4 ABILITY to apply good software engineering practices on given case study.

Excellent

(A+, A, A-) High

Good

(B+, B, B-) Significant

Fair

(C+, C, C-) Moderate

Marginal

(D) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Quiz (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion

- 1.1 ABILITY to describe, analyse and apply software engineering processes and techniques.

Excellent

(A+, A, A-) High

Good

(B+, B) Significant

Marginal

(B-, C+, C) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Research Paper Presentation (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion

- 2.1 ABILITY to explain and present technical contents in software engineering research papers.

Excellent

(A+, A, A-) High

Good

(B+, B) Significant

Marginal

(B-, C+, C) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Project (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion

3.1 ABILITY to apply software engineering techniques and good practices for actual software development.

3.2 ABILITY to use project management skills in team project.

3.3 ABILITY to report in an organised and logical way. All works are professional presented. All sources are correctly and thoroughly documented.

Excellent

(A+, A, A-) High

Good

(B+, B) Significant

Marginal

(B-, C+, C) Basic

Failure

(F) Not even reaching marginal levels

Assessment Task

Examination (for students admitted from Semester A 2022/23 to Summer Term 2024)

Criterion

4.1 ABILITY to explain software development processes and compare process models.

4.2 ABILITY to apply software analysis and design techniques.

4.3 ABILITY to apply project management techniques on given case study.

4.4 ABILITY to apply good software engineering practices on given case study.

Excellent

(A+, A, A-) High

Good

(B+, B) Significant

Marginal

(B-, C+, C) Basic

Failure

(F) Not even reaching marginal levels

Part III Other Information**Keyword Syllabus**

Overview of the software engineering discipline. Software engineering process models and trends. Software engineering standards, techniques, best practices, and guidelines. Software project management.

Syllabus;

1. Overview of the Software Engineering Discipline

History and overview of the software engineering discipline. Major roles, issues and problems. Current trends and directions.

2. Software Engineering Processes and Techniques

Overview of different software engineering process models, such as structured analysis and design, object-oriented analysis and design, agile methodologies, coding, maintenance, and testing, and their trends. Contrasting and comparing the different models and techniques. The individual processes within the process models (such as requirements, implementation, testing, etc.), their roles, issues, deliverables (both diagrams, documents and software) quality management and project management.

3. Software Engineering Standards, Best Practices, and Guidelines

Overview of different SE-related standards, best practices, and guidelines, such as those provided by IEEE, ACM, SEI, etc.

4. Software Project Management

Overview of project management concepts as they relate to software engineering, such as those outlined by PMI (and its PMIBOK), for example scope, schedule development, costing and quality management.

Reading List**Compulsory Readings**

Title	
1	There is no textbook for this course. Classic and latest software engineering research papers will be discussed in the lectures and tutorial sessions.

Additional Readings

Title	
1	Software Engineering Institute: http://www.sei.cmu.edu/
2	Project Management Institute: http://www.pmi.org/
3	IEEE SE Online: http://www.computer.org/portal/site/seportal/
4	IEEE SE Standards: http://standards.ieee.org/software/
5	IEEE/ACM SE Code of Ethics and Professional Practice: http://www.acm.org/about/se-code