The Hong Kong University of Science and Technology

UG Course Syllabus (Fall 2025-26)

[Course Title] Software Engineering

[Course Code] COMP3111

[No. of Credits] 4

[Any pre-/co-requisites] Prerequisite(s): COMP 2012 OR COMP 2012H. Exclusion(s): COMP 3111H, ISOM 3210

Name: Kenneth Wai-Ting LEUNG

Email: kwtleung@ust.hk

Office Hours: By appointment via email

Course Description

Methods and tools for planning, designing, implementing, validating, and maintaining large software systems. Project work to build a software system as a team, using appropriate software engineering tools and techniques.

Mode of Delivery: [BLD] Blended learning

Intended Learning Outcomes (ILOs)

By the end of this course, students should be able to:

1. Apply appropriate modeling techniques to design the software for an application of medium complexity: specify, model and document the data and processing requirements for an application using a modeling language and implement a disciplined software design for an application.
2. Apply appropriate software engineering techniques to implement an application of medium complexity: transform requirements for an application, expressed in a modeling language, into a working software system, select appropriate algorithms to implement the stated requirements and design and implement a variety of tests to validate that a software system meets its stated requirements.
3. Function effectively as a member of a software development team: organize, manage and participate in a small software development team and plan and schedule the activities involved in developing software for a medium-sized application.
4. Manage a software development environment by selecting appropriate software components to implement the stated requirements for a software system and configuring and managing a software development environment.

Assessment and Grading

This course will be assessed using criterion-referencing and grades will not be assigned using a curve. Detailed rubrics for each assignment are provided below, outlining the criteria used for evaluation.

Assessments:

Assessment Task	Contribution to Overall Course grade (%)	Due date
Pre-Lecture Quizzes	5%	Before each lecture*
In-Class Exercises	5%	Immediately after each class*
Labs	10%	Immediately after each lab*
Midterm	20%	Week8-Week10*
Group Project	20%	By the end of the semester*
Final Examination	40%	Final Exam Period*

* Assessment marks for individual assessed tasks will be released within two weeks of the due date.

Mapping of Course ILOs to Assessment Tasks

Assessed Task	Mapped ILOs	Explanation
Pre-Lecture Quizzes	ILO4	**ILO4:** Students are assessed on their understanding of appropriate software components, version control, continuous integration, and other foundational concepts in managing a development environment.
In-Class Exercises	ILO4	**ILO4:** Reinforces practical skills for configuring and managing real-world software development environments in tools like GitHub, Docker, and IDEs.
Labs	ILO4	**ILO4:** Students gain experience in setting up, configuring, and managing the software environments necessary for different stages of software development (e.g., testing frameworks, CI pipelines).
Midterm	ILO4	**ILO4:** Tests students' ability to analyze, select, and explain the configuration of development tools and software components in a given scenario.
Group Project	ILO1, ILO2, ILO3, ILO4	ILO1: Students work together to analyze requirements, create system models (e.g., UML), and document their software design using formal modeling techniques. ILO2: Students implement the designed system by transforming models into code, selecting and applying appropriate algorithms, and developing comprehensive test suites. ILO3: Students plan and execute the project collaboratively, demonstrating effective teamwork, task scheduling,

		and use of collaborative tools like version control. ILO4: While not the primary focus, students will also set up and manage the development environment necessary for team-based project execution.
Final Examination	ILO4	**ILO4:** Evaluates students' understanding of best practices for selecting, configuring, and maintaining development environments and tools throughout the software lifecycle.

Grading Rubrics

A detailed marking scheme will be prepared for each assignment and exam to clearly outline the evaluation criteria.

Final Grade Descriptors:

Grades	Short Description	Elaboration on subject grading description
A	Excellent Performance	Exhibits exemplary mastery of software engineering principles and practices. Designs and documents software systems with precision and creativity (ILO1). Implements solutions that are efficient, robust, and fully tested, including advanced use of appropriate algorithms (ILO2). Demonstrates outstanding teamwork, leadership, and collaboration in project planning and execution (ILO3). Configures and manages development environments effectively, showing strong command of tools such as version control, CI/CD pipelines, and testing frameworks (ILO4).
B	Good Performance	Shows strong understanding of software engineering concepts with only minor errors or lapses. Produces well-structured system models and generally complete documentation (ILO1). Implements functional solutions with appropriate algorithms and testing, though with occasional inefficiencies or missing details (ILO2). Works well in a team, contributing reliably to project goals (ILO3). Manages development environments competently, with only limited need for guidance (ILO4).
C	Satisfactory Performance	Demonstrates adequate understanding of software engineering concepts but with noticeable gaps or inconsistencies. Produces system designs that are functional but may lack clarity, completeness, or rigor (ILO1). Implements software with partial correctness and limited testing (ILO2). Participates in teamwork but with uneven contributions or difficulties in coordination (ILO3). Configures development environments at a basic level, though with errors or omissions (ILO4).
D	Marginal Pass	Displays limited understanding of software engineering principles, with frequent errors in system modeling and design (ILO1). Implementation shows significant flaws or incomplete functionality, with minimal testing (ILO2). Teamwork contributions are weak, inconsistent, or minimally effective (ILO3). Struggles with configuring and managing development tools and environments, requiring substantial support (ILO4).

F	Fail	Fails to demonstrate understanding of core software engineering concepts, with pervasive errors or missing work (ILO1). Unable to produce a functioning implementation or apply appropriate algorithms (ILO2). Does not contribute meaningfully to teamwork or project development (ILO3). Incapable of managing software tools and development environments at the required level (ILO4).

Course AI Policy

The use of generative AI tools is allowed for the course project only. Students must clearly document how AI tools are used, and remain responsible for the correctness, originality, and integrity of their submissions. AI tools are not permitted in quizzes, labs, midterm, or the final examination.

Communication and Feedback

Assessment marks for individual assessed tasks will be communicated via Canvas within two weeks of submission. Feedback on assignments will include [specific details, e.g., strengths, areas for improvement]. Students who have further questions about the feedback including marks should consult the instructor within five working days after the feedback is received.

Resubmission Policy

No resubmission is allowed for any assessed tasks. Students are expected to submit all work by the specified deadlines.

Required Texts and Materials

N/A

Academic Integrity

Students are expected to adhere to the university's academic integrity policy. Students are expected to uphold HKUST's Academic Honor Code and to maintain the highest standards of academic integrity. The University has zero tolerance of academic misconduct. Please refer to Academic Integrity | HKUST – Academic Registry for the University's definition of plagiarism and ways to avoid cheating and plagiarism.

[Optional] Additional Resources

N/A