



Course Specification

(Bachelor)

Course Title: **Foundations of Software Engineering**

Course Code: **SE1201**

Program:

BSc in Software Engineering

Bachelor of Data Science

BSc in Artificial Intelligence

BSc in Computer Science

BSc in Human Computer Interaction

Department: **Software Engineering**

College: **College of Computing**

Institution: **Umm Al Qura University**

Version: **1.0**

Last Revision Date: **22/04/2025**



Table of Contents

A. General information about the course:	3
B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods	4
C. Course Content	8
D. Students Assessment Activities	10
E. Learning Resources and Facilities	10
F. Assessment of Course Quality	11
G. Specification Approval	11



A. General information about the course:

1. Course Identification

1. Credit hours: (3)

2. Course type

- A. University College Department Track Others
- B. Required Elective

3. Level/year at which this course is offered:

BSc in Software Engineering AND BSc in Human Computer Interaction - (1nd year/ 2th level)
BSc in Computer Science AND BSc in Artificial Intelligence - (2nd year/ 4th level)
Bachelor of Data Science (3rd year/ 5th level)

4. Course General Description:

Foundations of Software Engineering teaches core principles, methodologies, and practices in software engineering. The course includes software processes, agile development, requirements engineering, system modeling, and key aspects of design, implementation, testing, and maintenance.

5. Pre-requirements for this course (if any):

BSc in Software Engineering; BSc in Human Computer Interaction; BSc in Computer Science AND Bachelor of Data Science:

N/A

BSc in Artificial Intelligence:

DS2201-Introduction to Database

6. Co-requisites for this course (if any):

N/A

7. Course Main Objective(s):

Upon successful completion of this course, you will be able to:

1. Describe the knowledge and skills essential for practicing software engineering and the professional challenges software engineers may encounter.
2. Apply software engineering principles such as separating concerns, abstraction, and incremental development to create reliable software.
3. Differentiate between various software development processes and methodologies.
4. Identify key activities and deliverables in the software development life cycle, including use case, class, and sequence diagrams.



5. Create UML diagrams for software analysis and design using object-oriented methodology.
6. Employ project management concepts to manage projects, personnel, and products effectively.
7. Utilize software engineering principles to build high-quality software systems.

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	45	100%
2	E-learning	0	0
3	Hybrid <ul style="list-style-type: none"> • Traditional classroom • E-learning 	0	0
4	Distance learning	0	0

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	45
2.	Laboratory/Studio	.
3.	Field	0
4.	Tutorial	0
5.	Others (specify)	0
Total		45

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods (BSc in Software Engineering)

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			



Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.1	Define key concepts, principles, and terminologies of software engineering.	K1	Lecture, Exercise	Quiz, Exams, Assignments
1.2	Explain various software development life cycles, including traditional and agile approaches.	K1	Lecture, Case Studies, Discussions	Quiz, Exams, Assignments
2.0	Skills			
2.1	Apply software process models and methodologies to solve software engineering problems.	S1	Group Discussion, Tutorials	Problem-Solving Tasks, Project
2.2	Develop and document software requirements using techniques such as use cases and user	S3	Group Projects, Tutorials	Assignments, Project
2.3	Design and implement basic software architecture and system models.	S2	Lecture, Group Discussion, Tutorials	Assignments, Project
3.0	Values, autonomy, and responsibility			
3.1	Demonstrate an understanding of ethical and professional practices in software development.	V1	Group Discussions, Guest Lectures	Reflection Essays on Ethical Scenarios.
3.2	Collaborate effectively within a team to achieve project goals in line with software engineering standards.	V2	Group Projects, Tutorials	Group Project

Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods (Bachelor of Data Science)

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			



Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.1	Define key concepts, principles, and terminologies of software engineering.	K1	Lecture, Exercise	Quiz, Exams, Assignments
1.2	Explain various software development life cycles, including traditional and agile approaches.	K1, K3	Lecture, Case Studies, Discussions	Quiz, Exams, Assignments
2.0	Skills			
2.1	Apply software process models and methodologies to solve software engineering problems.	S1	Group Discussion, Tutorials	Problem-Solving Tasks, Project
2.2	Develop and document software requirements using techniques such as use cases and user	S2	Group Projects, Tutorials	Assignments, Project
2.3	Design and implement basic software architecture and system models.	S4	Lecture, Group Discussion, Tutorials	Assignments, Project
3.0	Values, autonomy, and responsibility			
3.1	Demonstrate an understanding of ethical and professional practices in software development.	V2	Group Discussions, Guest Lectures	Reflection Essays on Ethical Scenarios.
3.2	Collaborate effectively within a team to achieve project goals in line with software engineering standards.	V3	Group Projects, Tutorials	Group Project

Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods (Bachelor of Science in Computer Science)

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			



Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.1	Define key concepts, principles, and terminologies of software engineering.	K1	Lecture	Quiz, Exams, Assignments
1.2	Explain various software development life cycles, including traditional and agile approaches.	K2	Lecture, Case Studies, Discussions	Quiz, Exams, Assignments
2.0	Skills			
2.1	Apply software process models and methodologies to solve software engineering problems.	S1	Group Discussion, Tutorials	Problem-Solving Tasks, Project
2.2	Develop and document software requirements using techniques such as use cases and user	S3	Group Projects, Tutorials	Assignments, Project
2.3	Design and implement basic software architecture and system models.	S2	Lecture, Group Discussion, Tutorials	Assignments, Project
3.0	Values, autonomy, and responsibility			
3.1	Demonstrate an understanding of ethical and professional practices in software development.	V2	Group Discussions, Guest Lectures	Reflection Essays on Ethical Scenarios.
3.2	Collaborate effectively within a team to achieve project goals in line with software engineering standards.	V3	Group Projects, Tutorials	Group Project

Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods (Bachelor of Science in Artificial Intelligence)

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			





Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.1	Define key concepts, principles, and terminologies of software engineering.	K1	Lecture, Exercise	Quiz, Exams, Assignments
1.2	Explain various software development life cycles, including traditional and agile approaches.	K2	Lecture, Case Studies, Discussions	Quiz, Exams, Assignments
2.0	Skills			
2.1	Apply software process models and methodologies to solve software engineering problems.	S1	Group Discussion, Tutorials	Problem-Solving Tasks, Project
2.2	Develop and document software requirements using techniques such as use cases and user	S2	Group Projects, Tutorials	Assignments, Project
2.3	Design and implement basic software architecture and system models.	S2	Lecture, Group Discussion, Tutorials	Assignments, Project
3.0	Values, autonomy, and responsibility			
3.1	Demonstrate an understanding of ethical and professional practices in software development.	V1	Group Discussions, Guest Lectures	Reflection Essays on Ethical Scenarios.
3.2	Collaborate effectively within a team to achieve project goals in line with software engineering standards.	V2	Group Projects, Tutorials	Group Project

Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods (BSc in Human Computer Interaction)

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			



Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.1	Define key concepts, principles, and terminologies of software engineering.	K1	Lecture, Exercise	Quiz, Exams, Assignments
1.2	Explain various software development life cycles, including traditional and agile approaches.	K1, K2	Lecture, Case Studies, Discussions	Quiz, Exams, Assignments
2.0	Skills			
2.1	Apply software process models and methodologies to solve software engineering problems.	S1	Group Discussion, Tutorials	Problem-Solving Tasks, Project
2.2	Develop and document software requirements using techniques such as use cases and user	S2, S3	Group Projects, Tutorials	Assignments, Project
2.3	Design and implement basic software architecture and system models.	S4	Lecture, Group Discussion, Tutorials	Assignments, Project
3.0	Values, autonomy, and responsibility			
3.1	Demonstrate an understanding of ethical and professional practices in software development.	V1	Group Discussions, Guest Lectures	Reflection Essays on Ethical Scenarios.
3.2	Collaborate effectively within a team to achieve project goals in line with software engineering standards.	S5	Group Projects, Tutorials	Group Project

C. Course Content

No	List of Topics	Contact Hours
1.	Introduction to Software Engineering	3
2.	Software Processes	6
3.	Agile Software Development	3
4.	Software Requirements Engineering	6
5.	Software Modeling (Context Diagram, Use Case Diagram, Domain Models, and Sequence Diagrams)	6
6.	Software Design and Architectural	6





7.	Software Testing	6
8.	Software Evolution	6
9.	Project Management	3
Total		45

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Quizzes	2-14	15
2.	Projects	2-14	15
3.	Assignments	2-14	10
4.	Mid Term	7	20
5.	Final Exam	16-17	40

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	<ul style="list-style-type: none"> Sommerville, I. (2015). <i>Software engineering</i> (10th ed.). Pearson.
Supportive References	<ul style="list-style-type: none"> Pressman, R. S., & Maxim, B. R. (2019). <i>Software engineering: A practitioner's approach</i> (9th ed.). McGraw-Hill Higher Education. Kim, G., Debois, P., Willis, J., & Humble, J. (2016). <i>The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations</i>. IT Revolution Press. Murphy, N., Beyer, B., Jones, C., & Petoff, J. (2016). <i>Site reliability engineering: How Google runs production systems</i>. O'Reilly Media.
Electronic Materials	
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Classroom





Items	Resources
Technology equipment (projector, smart board, software)	Projector
Other equipment (depending on the nature of the specialty)	

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Students	Direct, Indirect
Effectiveness of Students' assessment	Faculty, Peer reviewer	Direct, Indirect
Quality of learning resources	Faculty, Course coordinator	Direct, Indirect
The extent to which CLOs have been achieved	Course coordinator, Program management committee	Direct
Other		

Assessors (Students, Faculty, Program Leaders, Peer Reviewers, Others (specify))

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	SOFTWARE ENGINEERING DEPARTMENT COUNCIL
REFERENCE NO.	THE 17TH MEETING FOR THE ACADEMIC YEAR 1446H
DATE	22/04/2025

Council / Committee	Computer Science and Artificial Intelligence Department Council
Reference No.	THE 16TH MEETING FOR THE ACADEMIC YEAR 1446H
Date	22/04/2025

COUNCIL /COMMITTEE	Data Science Department Council
REFERENCE NO.	Meeting number 11, academic year 1446
DATE	22/4/2025

