



AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH (AIUB)

Faculty of Science and Technology (FST)

Department of Computer Science (CS)

Undergraduate Program

COURSE PLAN

Summer 2020-2021 SEMESTER

I. Course Core and Title

CSC 3114: Software Engineering

II. Credit

3 credit hours (2 hours of theory & 3 hours of lab class per week)

III. Nature

Core Course for CSE

IV. Prerequisite

CSC 2210: Object Oriented System Analysis and Design

V. Vision:

Our vision is to be the preeminent Department of Computer Science through creating recognized professionals who will provide innovative solutions by leveraging contemporary research methods and development techniques of computing that is in line with the national and global context.

VI. Mission:

The mission of the Department of Computer Science of AIUB is to educate students in a student-centric dynamic learning environment; to provide advanced facilities for conducting innovative research and development to meet the challenges of the modern era of computing, and to motivate them towards a life-long learning process.

VII - Course Description:

- Comprehend introduction to the modern study of software engineering.
- Discuss the present software engineering practices.
- Discuss various process models used software engineering
- Describe requirements engineering and design process.
- Comprehend the technologies used in coding and testing.
- Discuss the software project management and planning

VIII – Course outcomes (CO) Matrix:

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

COs *	Description	Level of Domain**				PO Assessed***
		C	P	A	S	
CO1	Describe for impact of software engineering models over various context of software development	2			CT	7.1
CO2	Discuss the software engineering models for sustainability software development in a dynamic environment	2			CT	7.2
CO3	Choose appropriate software engineering model in a software development environment		5		CT	6.1
CO4	Explain the software project management roles and their skills in a team environment		5		CT	6.2

C: Cognitive; P: Psychomotor; A: Affective; S: Soft-skills (CT: Critical Thinking, TS: Teamwork)

* CO assessment method and rubric of COs assessment is provided in Appendix section

** The numbers under the 'Level of Domain' columns represent the level of Bloom's Taxonomy each CO corresponds to.

*** The numbers under the 'PO Assessed' column represent the PO each CO corresponds to.

IX – Topics to be covered in the class:

TOPICS	Learning Objective(s)	Time Frame	Suggested Activities	Teaching Strategy(s)	CO mapped
Introduction to Software Engineering - Software & Software Engineering	Knowing Mission & Vision of AIUB and course outline Understand various building blocks of software and engineering	Week 1	Lecture: necessary explanation of the software and software engineering	Lecture notes, reallife examples, question-answer	
Traditional Software Development Process Models	Understanding the different traditional process models in software engineering	Week 2	Lecture: necessary explanation on phases of traditional process models	Lecture notes, reallife examples, question-answer	CO1 CO2
Agile Development	Understand and appreciate the concepts of Agile methods in software engineering practice	Week 2	Lecture: necessary explanation on principles of agile development	Lecture notes, reallife examples, question-answer	CO1 CO2
Extreme Programming (XP)	Understand the values, phases, roles, and practices of XP in software engineering	Week 3	Lecture: necessary explanation on XP process in software development	Lecture notes, reallife examples, question-answer	CO3 CO4
SCRUM	Understand the phases, roles, and practices of SCRUM in software engineering	Week 3 and 4	Lecture: necessary explanation on SCRUM process in software development	Lecture notes, reallife examples, question-answer	CO3 CO4
Dynamic System Development Method (DSDM)	Understand the phases, roles, and practices of DSDM in software engineering	Week 4 and 5	Lecture: necessary explanation on DSDM process in software development	Lecture notes, reallife examples, question-answer	CO3 CO4
Feature Driven Development (FDD)	Understand the process, roles and, practices of FDD in software engineering	Week 5	Lecture: necessary explanation on FDD process in software development	Lecture notes, reallife examples, question-answer	CO3 CO4
Requirements Engineering	Understand the engineering process of requirements elicitation, analysis, modelling elements, specification and validation methods	Week 6	Lecture: necessary explanation on the requirements engineering processes	Lecture notes, reallife examples, question-answer	
Midterm Exam Week Week 7					

Design Concepts and User Interface Design	Understand various principles of software and user interface design	Week 8	Lecture: necessary explanation on design principles of software and user interface	Lecture notes, reallife examples, question-answer	
Testing Strategies	Understand various testing techniques and debugging strategies	Week 9	Lecture: necessary explanation on testing techniques and debugging	Lecture notes, reallife examples, question-answer	
Software Quality Attributes	Understand various quality attributes and their inter relations, trade-offs in software operations	Week 10	Lecture: necessary explanation on software quality attributes	Lecture notes, reallife examples, question-answer	
Product Metrics	Understand various Software Metrics and their usage	Week 11	Lecture: necessary explanation on software product metrics	Lecture notes, reallife examples, question-answer	
Software Configuration Management	Understand Software Configuration Management principles and version controlling	Week 12	Lecture: necessary explanation on software configuration management	Lecture notes, reallife examples, question-answer	
Estimation for Software Projects	Understand various conventional software project estimation techniques	Week 13	Lecture: necessary explanation on software effort estimation	Lecture notes, reallife examples, question-answer	
Project Scheduling	Understand various conventional software project scheduling techniques	Week 13	Lecture: necessary explanation on software project scheduling	Lecture notes, reallife examples, question-answer	
Risk Management	Understand various Risk management strategy	Week 13	Lecture: necessary explanation on risk management in project development	Lecture notes, reallife examples, question-answer	
Final term Exam Week Week 14					

* The faculty reserves the right to change, amend, add or delete any of the contents.

XI- Course Requirements

1. Student Attendance

All students are expected to attend all scheduled classes as well as counselling, and to read all assigned chapters/materials before coming to class. At least 80% class attendance is mandatory to pass the course. If there is any assignment given to the students, they have to submit it before the deadline decided by the course teacher

2. Class Participation & Peer Evaluation

Students are expected to participate actively in the class. Your contribution towards your team will be counted too.

3. Quiz & Exam

For both terms, there will be at least 2 quizzes based on the theoretical knowledge and conceptual understanding of the topic covered discussed in the classes.

4. Assignment/Projects

Submit report based on the given software engineering related problems in the class. Assignment report should be presented by the dateline assigned.

XII – Evaluation & Grading System

The following grading system will be strictly followed in this class

Marking system for Theory Classes (Midterm and Final term)		Letter	Grade Point	Numerical %
Quiz(s):	40%	A+	4.00	90-100
Class Attendance and Performance:	10%	A	3.75	85-89.99
Midterm/Final term exam:	50%	B+	3.50	80-84.99
Total	100%	B	3.25	75-79.99
Final Grade/ Grand Total		C+	3.00	70-74.99
Quiz(s):	40%	C	2.75	65-69.99
Final Term:	40%	D+	2.50	60-65.99
Assignment/Projects	20%	D	2.25	50-59.99
Total	100%	F	0.00	<50(Failed)

The evaluation system will be strictly followed as par the AIUB grading policy.

XIII – Teaching Methods

Maximum topics will be covered from the textbook. For the rest of the topics, reference books will be followed. Some Class notes will be uploaded on the web. White board will be used for most of the time.

For some cases, multimedia projector will be used for the convenience of the students.

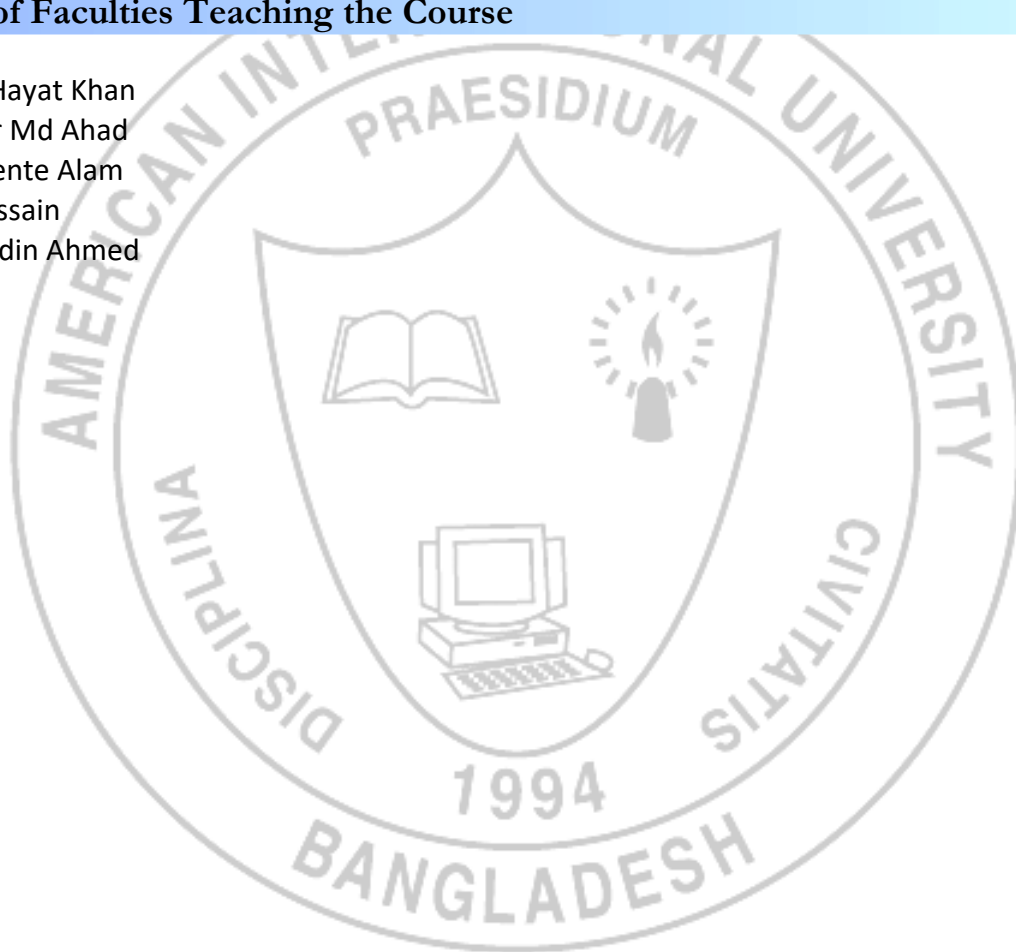
Students must study up to the last lecture before coming to the class and it is suggested that they should go through the relevant chapter before coming to the class. Just being present in the class is not enough- students must participate in classroom discussions.

XIV – Textbook/ References

- 1) Software Engineering: A Practitioner's Approach, Seventh Edition, Roger S. Pressman
- 2) Software Engineering, Sommerville
- 3) An Integrated Approach to Software Engineering, Pankaj Jalote
- 4) Object Oriented Software Engineering, Ivar Jacobson, Magnus Christerson, Patrik Jonsson, Gunnar Overgaard
- 5) Systems Analysis and Design: An Object-Oriented Approach with UML, 5th Edition, Alan Dennis
- 6) The Art of Computer Programming, The, Volumes 1-3 Boxed Set (2nd Edition), Donald E. Knuth
- 7) Component Software: Beyond Object-Oriented Programming, Clemens Szyperski
- 8) Practices of an Agile Developer: Working in the Real World, Venkat Subramaniam, Andy Hunt
- 9) Code Complete: A Practical Handbook of Software Construction, Steve McConnell
- 10) Lectures will be provided online at the course website weekly.

XV - List of Faculties Teaching the Course

Dr. Razib Hayat Khan
Dr. Taimur Md Ahad
Farzana Bente Alam
Sazzad Hossain
Raihan Uddin Ahmed



XVI – Verification:

Prepared by : ----- Dr. Razib Hayat Khan <i>Course Convener</i> Date:.....	Moderated by : ----- Mahmudul Hasan <i>Point Of Contact</i> <i>OBE Implementation Committee for CS</i> Date:.....	
Checked by: ----- Dr. Mahbub Chowdhury Mishu <i>Head,</i> <i>Department of Computer Science</i> Date:.....	Certified by: ----- Dr. Dip Nandi <i>Director,</i> <i>Faculty of Science & Information Technology</i> Date:.....	Approved by: ----- Mr. Mashiour Rahman <i>Associate Dean,</i> <i>Faculty of Science & Information Technology</i> Date:.....

APPENDIX

Program Outcomes (POs)

PO7	Environment and sustainability
7.1	Understand the impact of professional engineering solutions in societal and environmental contexts
7.2	Demonstrate the knowledge of and need for sustainable development.
PO6	The engineer and society
6.1	Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues
6.2	Assess the consequent responsibilities relevant to professional engineering practice.

Mapping of CO - Assessment Method and Rubric

The mapping between Course Outcome(s) (COs) and The Selected Assessment method(s) and the mapping between Assessment method(s) and Evaluation Rubric(s) is shown below:

COs	Description	POs	Learning Domain	Assessment Method	Assessment Rubric
CO1	<i>Describe</i> the impact of software engineering models over various context of software development	7.1	Cognitive	Quiz	Rubric for Quiz
CO2	<i>Discuss</i> the software engineering models for sustainable software development in a dynamic environment	7.2	Cognitive	Quiz	Rubric for Quiz
CO3	<i>Choose</i> appropriate software engineering model in a software development environment	6.1	Psychomotor	Project	Rubric for Project
CO4	<i>Explain</i> the roles and their responsibilities in the software project management activities	6.2	Psychomotor	Project	Rubric for Project

Rubric for Quiz Assessment (CO1 & CO2)

Marking Criteria	Marks Distribution (Maximum 5X3=15)	Acquired Marks
------------------	-------------------------------------	----------------

	Inadequate (1-2)	Satisfactory (3)	Good (4)	Excellent (5)	
Definition	Student vaguely define the terms or the concept of the s/w engineering models	Definition provided with partial relevance to the model	Correctly define the model with minor missing of model details.	Comprehensively defines the model.	
Logical Argument	No logical arguments / explanation supporting the definition.	Offers lossy related arguments.	Strong argument /explanation offered.	Comprehensive argument presented to clarify the concept.	
Relevant Example	Ambiguous representative example.	Correctly identify / indicate towards real-life example.	Real-life example is strongly connected towards the definition.	Comprehensively defend with real life example.	
Acquired Marks:					
CO Pass / Fail:					



Rubric for Project Assessment (CO3)

Criteria	Marks distribution (Max 4X5= 20)				Acquired Marks
	Inadequate (1-2)	Satisfactory (3)	Good (4)	Excellent (5)	
Content Knowledge	Student does not have grasp of information and cannot answer the questions about subject.	Student is uncomfortable with information and is able to answer only basic questions.	Student is at ease with content but fails to elaborate.	Student demonstrates full knowledge (more than required) with explanations and elaboration.	
Argumentation	Does not articulate a position or argument of choosing appropriate model	Articulates a position or argument for choosing models that is unfocused or ambiguous	Articulates a position or argument of choosing models that is incomplete or limited in scope	Clearly articulates a position or argument for the choosing software engineering models	
Evidence of Argumentation	Does not present any evidence to support the arguments for the choice of the model	Presents incomplete/vague evidence to support argument for model choice	Does not present enough evidence to support the argument for the choice of the model	Presents sufficient amount of evidence to support argument for the model selection	
Completeness, Spelling, grammar and Organization of the Answer	Answer is not complete and Several errors in spelling and grammar. Present a Confusing organization of concepts, supporting arguments, and real-life example. Sentences rambling, and details are repeated	Some errors in spelling and grammar. Some problems of organizing the answer in a logical order of defining, elaborating, and providing real-life examples	Few errors in spelling and grammar. Presents most of the details in a logical flow of organization in definition, details, and example	Answer is complete and No errors in spelling and grammar. Consistently presents a logical and effective organization of definition, details, and real-life example of the topic	
Acquired marks:					
CO Pass / Fail:					

Rubric for Project Assessment (CO4)

Criteria	Marks distribution (Max 4X5= 20)				Acquired Marks
	Inadequate (1-2)	Satisfactory (3)	Good (4)	Excellent (5)	
Background Analysis	No background information is given; project goals and benefits are missing.	Insufficient background information is given; project goals and benefits are poorly stated	Sufficient background information is given; the purpose and goals of the project are explained.	Thorough and relevant background information is given; project goals are clear and easy to identify.	
Role identification	Does not identify any roles in the project management activities	Identify few roles in the project management	Identify most of the roles in the project management	Identify all of the roles in the project management activities	
Responsibility Allocation	The project has poor project management plans for assigning the responsibilities	Some of the roles are left alone with any project responsibilities	Few of the roles are left alone with any project responsibilities	Well planned project with proper resource allocation	
Completeness, Spelling, grammar and Organization of the Answer	Answer is not complete and Several errors in spelling and grammar. Present a Confusing organization of concepts, supporting arguments, and real-life example. Sentences rambling, and details are repeated	Some errors in spelling and grammar. Some problems of organizing the answer in a logical order of defining, elaborating, and providing real-life examples	Few errors in spelling and grammar. Presents most of the details in a logical flow of organization in definition, details, and example	Answer is complete and No errors in spelling and grammar. Consistently presents a logical and effective organization of definition, details, and real-life example of the topic	
Acquired marks:					
CO Pass / Fail:					