



**Subject:** Software Engineering -BSCIT14313

**Type of course:** Major Core

**Prerequisite:** Basic knowledge of Software Development process with involves Designing, Developing, and testing new technologies to Solve Real-world problems.

**Rationale:**

Software Engineering is the important lessons of software engineering only apply to large scale Software development the skills needed to develop such systems have very little to do with the type Of programming.

In fact, software engineering has to do with programming at all, Software Engineering is about the Discipline needed to develop high quality software that can be understood, maintained and adapted Over long periods of time by many different people.

An understanding of what software quality really means is central to understanding what software by attempting to understand, assess, and modify one another's programs, students will gain insight Into the nature of software quality, and why ability to program is not sufficient for the construction Of high-quality software.

**Teaching and Examination Scheme:**

Teaching Scheme				Credits	Examination Marks					Total Marks		
CI	T	P	C		SEE		CCE					
					Theory	Practical	MSE	LWA	ALA			
3	0	2	4		75	25	30	20	50	200		

Legends: CI-Class Room Instructions; T – Tutorial; P - Practical; C – Credit; SEE - Semester End Evaluation; MSE- Mid Semester Examination; LWA - Lab Work Assessment; V – Viva voce; CCE- Continuous and Comprehensive Evaluation; ALA- Active Learning Activities.



**CourseContent:**

Sr. No	Course content	Hrs	% Weightage
1	<p><b>Introduction to Software Engineering:</b></p> <p>Definition, Scope, and Importance of Software Engineering, The Evolving Role of Software and the Software Engineer, Software Development Life Cycle (SDLC): Phases, Activities, and Deliverables, Software Process Models (Incremental Model, Spiral Model, V-Model, Agile Models).</p>	12	25%
2	<p><b>Software Requirements Engineering:</b></p> <p>Importance of Design: UMLDiagram and types, Introduction to Programming Frameworks, Role of Requirements software engineering, Validation and Verification</p>	10	25%
3	<p><b>Software Design and Architecture</b></p> <p>Principles of Software Design, System Design and Architecture design (Layered Architecture, Client-Server Architecture, Microservices Architecture, Data Design and Data Flow Diagrams (DFDs)</p>	08	20%
4	<p><b>Software coding- Testing and Quality Assurance</b></p> <p>Coding Standard and coding Guidelines, Code Review, Testing, Testing Fundamentals, level of testing, types of testing. Quality Metrics, QA Standards and Models, Risk Management.</p>	08	15%
5	<p><b>Current trends in Software Engineering-Agile Development</b></p> <p>Agile Methodologies, Principles, History and Evolution of Agile, Scrum Framework, Scrum Roles, FDD Overview, and process.</p>	07	15%

**Continuous Assessment:**

(For each activity maximum-minimum range is 5 to 10 marks)

Sr. No	Active Learning Activities	Marks
1	<p><b>The Art of Effective Presenting</b></p> <p>Create presentation on given topic, prepare its video and upload it on GMIU</p>	10



	Web portal.	
2	<b>Diagram Creation</b> Students will create Advance UML class diagram in chart paper on given topic and upload it on GMIU web portal.	10
3	<b>Preparing Test Case:</b> Students have to create list of test cases on given scenarios and upload that document on GMIU Web portal.	10
4	<b>Creation with Agile:</b> Students will perform case study using Agile Software development and upload its documentation on GMIU web portal.	10
5	<b>SRS:</b> Students have to prepare an SRS and upload it on GMIU Web portal.	10
Total		50

**Suggested Specification table with Marks (Theory):75**

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage	25%	45%	15%	15%	0	0

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

**Course Outcome:**

After learning the course the students should be able to:	
CO1	Understand the concept of software engineering, software process model, requirements of software engineering and how to manage client's requirements
CO2	Know Software engineering diagramsconcept
CO3	Know design concept and user interface
CO4	Know coding concept and designing interface
CO5	Understand agile methodology and scrum development processing



**List of Practical**

Sr. No	Descriptions	Hrs
1	Study the complete Software Development Life Cycle (SDLC) and analyze various activities conducted as a part of various Phases.	4
2	Prepare Use Case Diagram for Library Management System.	4
3	Prepare DFD diagram for Banking System.	2
4	Prepare ER diagram for a College Management System.	2
5	Develop Activity diagram for Restaurant Management System.	2
6	Develop data designs one to many entities' relationship diagram	2
7	Develop a class diagram of E-commerce website.	2
8	Develop test case for Flight Control System.	4
9	Develop scrum backlogs of your IS-2 subject Project.	2
10	Development of Software Requirements Specification (SRS).	6
		30

**Instructional Method:**

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory

The internal evaluation will be done on the basis of Active Learning Assignment

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.



**Reference Books:**

- [1] Waman S. Jawadekar: Software Engineering— Principles and Practices: TMGH Publication
- [2] Roger S Pressman: Software Engineering- Software Engineering: A Practitioner Approach
- [3] Ian Sommerville: Software Engineering- Software Engineering: Seventh Edition
- [4] A. A. Puntambekar: Software Engineering
- [5] pankajJalote Software Engineering: A Precise Approach

