MCA Semester I

Subject Name: Agile Methodologies and Unified Modeling**Subject Code:** MTCA13105
Type of course: Professional Core Course

Prerequisite (if any): None

List of Courses where this course will be prerequisite: Software Testing, Software Quality Assurance, and Agile Project Development

Rationale:

- To Study technical and managerial aspects of Software Development.
- To Study Development Models andAgile Software Developmentprocess Model.
- To Study Requirement Gathering and SRS Documentation.
- To Study Design Function Oriented and Object Oriented Design Approach.
- To Study Testing Strategies and Testing Tactics.
- To Study UML for Object Oriented Modeling and Design.
- To Study about Git and able to create and merge gitbranches.

Teaching and Examination Scheme:

TEACHING SCHEME				Theory Marks			Practical Marks		Total
L	T	P	C	TEE	CA1	CA2	TEP	CA3	
3	0	0	3	60	25	15	0	0	100

CA1: Continuous Assessment (assignments/projects/open book tests/closed book tests **CA2:** Sincerity in attending classes/class tests/ timely submissions of assignments/self-learning attitude/solving advanced problems **TEE:** Term End Examination **TEP:** Term End Practical Exam (Performance and viva on practical skills learned in course) **CA3:** Regular submission of Lab work/Quality of work submitted/Active participation in lab sessions/viva on practical skills learned in course

Content:

Sr. No.	Topics	Teaching Hrs.	Module Weightage
1	**Introduction to Software Engineering** Introduction to Software engineering, Software Development Life Cycles, Requirements Engineering, Design and Architectural Engineering, Design Model, Object Oriented Analysis and Design.	7	15%
2	**Testing** Introduction to Software Testing, Principals of Software Testing, Verification & Validation, Quality Assurance Vs. Quality Control, Introduction to STLC and V Models, Types of Testing, Tools, Testing strategies, White box testing and Black box testing.	7	15%
3	**Agile Development** Introduction to Agile development model, Benefits of Agile model, Agility and Agile Process, Agile Process model, Adaptive Software Development, Dynamic Systems Development Method, Crystal, Feature Driven Development and Agile Modeling	8	20%
4	**Scrum and Extreme Programming (XP)** Scrum and Self-Organizing, Scrum Planning and Collective Commitment, Sprint, Retrospective, XP and Embracing change, Simplicity and Incremental Design, Kanban, Story Grooming	7	15%

5	**Basics of Object Oriented Modeling and Design, UML, Advanced Structural Modeling** Introduction to Object Oriented Software Development Life Cycle, Advantages of Object Oriented Methodology, Why We Model, Introduction to UML, Classes, Relationships,Common Mechanisms, Diagrams and Class Diagrams, Advanced Classes, Advanced Relationships, Instances, Object Diagrams.	**9**	**20%**
6	**Basic and Advanced Behavioral Modeling** Interactions, Use Cases, Use Case Diagrams, Interaction Diagrams,Activity Diagrams, Events and Signals, State Machines, and Statechart Diagrams.	**7**	**15%**

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
20	20	15	15	15	15

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

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.

Reference Books:

Sr no	**Title of book /article**	**Author(s)**	**Publisher and details like ISBN**	**Year of publication**	**Publication Edition**
1	Software Engineering- A practitioner's	Roger S.Pressman,	McGraw-Hill International	2015	8th Edition

	Approach				
2	Software Engineering – A Precise Approach	PankajJalote	Wiley	2010	2nd Edition
3	Software Engineering	Ian Sommervill e	Pearson education Asia	2011	9th Edition
4.	The Unified Modeling Language User Guide	Grady Booch, James Rumbaugh, Ivar Jacobson	Pearson Education	2009	1st Edition
5.	Learning Agile	Andrew Stellman, Jennifer Greene	O'Reilly, Shorff Publishers and Distributers Pvt. Ltd.	2019	1st Edition Fourth Indian Reprint
6.	Git Pocket Guide: A Working Introduction	Richard E. Silverman	O'Reilly Media	2013	1st Edition

Course Outcomes:

Sr. No.	CO Statement After learning this subject, students will be able to	Marks % weightage
CO-1	Prepare SRS (Software Requirement Specification) document and apply Function Oriented and Object Oriented Design approaches to design the software.	15%

CO-2	Analyse and apply relevant Software Testing strategies and techniques on a given system.	15%
CO-3	Study and utilize Agile Software Development Process.	35%
CO-4	Identify and develop UML models for Object-Oriented Modeling and Design.	35%

Mapping with POs:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO-1	3	3	0	2	0	1	3	3	3	0	0	3			
CO-2	3	3	0	1	2	1	3	2	3	0	0	3			
CO-3	3	3	0	1	2	1	3	2	3	0	0	3			
CO-4	3	3	0	1	0	1	3	3	3	0	0	3			
Rationale*															

Rationale*: Explaining why it is matching this particular program outcome

List of Open learning website:

- http://www.rspa.com/spi/
- https://www.atlassian.com/agile/
- https://www.scrum.org/
- https://git-scm.com/docs/gittutorial
- https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners

List of Open Source Software:

- https://staruml.io/