



SILVER OAK UNIVERSITY

College of Technology

Bachelor of Technology

Information Technology

Course Name: Software Engineering

Course Code:1010043336

Semester:5th

Prerequisite:

Fundamental of UML

Objective:

1. To study Software Development Life Cycle, Development models and Agile Software development.
2. To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
3. To discuss various software testing issues and solutions in software unit test, integration, regression, and system testing.
4. To gain the techniques and skills on how to use modern software testing tools to support software testing projects.

Teaching Scheme:

Teaching Scheme				
L	T	P	Contact Hours	Credit
3	0	2	5	4

Contents:

Unit	Topics	Teaching Hours	Weightage %
1	INTRODUCTION TO SOFTWARE ENGINEERING: Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models, Agile Process Model, Component-Based Development, Process, Product and Process.	04	12
2	AGILE SOFTWARE DEVELOPMENT: Agility and Agile Process Model, Extreme Programming, Scrum ,Kanban, XP, Site Reliability Engineering (SRE), Roles and Types of Standards, ISO 12207: Life Cycle Standard IEEE Standards for Software Engineering Processes and Specifications .	05	10
3	SOFTWARE REQUIREMENT MODELING AND SPECIFICATION: Requirements Development Methodology, Specifying Requirements (SRS), Eliciting Accurate Requirements	05	15

	,Documenting Business Requirements, Defining User Requirements, Validating Requirements, Achieving Requirements Traceability, Managing Changing Requirements, Reviews, Walkthroughs, and Inspections, Requirements Modeling, Agile Requirements Engineering.		
4	SOFTWARE ANALYSIS AND DESIGN: Roles of Analysis and Design, Design Concepts and Design Principal, Architectural Design, Component Level Design (Function Oriented Design, Object Oriented Design) (MS Visio Tool), User Interface Design, Web Application Design.	05	12
5	SOFTWARE PROJECT MANAGEMENT: W5HH of Project Management, Software Metrics (Process, Product and Project Metrics), Software Measurement: Metrics for Software Cost and Effort estimations, Software Project Estimation, Software Project Planning (MS Project Tool), Project Scheduling & Tracking, Risk Analysis & Management (Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation). Challenges and solution for PM activities, CMM	05	15
6	SOFTWARE CODING & TESTING: Coding Standard and coding Guidelines, Code Review, Software Documentation, STLC, Testing Strategies, Testing Techniques and Test Case, Test Suites Design, Testing Conventional Applications, Testing Object Oriented Applications, Testing Web and Mobile Applications .	05	10
7	SOFTWARE QUALITY ASSURANCE, SOFTWARE MAINTENANCE AND SOFTWARE CONFIGURATION MANAGEMENT: Software Quality Concepts, Types of Maintenance, Software Configuration Management (SCM), Software Quality Assurance (SQA) , Software Quality and Agile Methods , Automated and Manual Functional Testing , Acceptance testing , User inter facetesting (HTTPUnit, Canoo), Performance testing , Software Metrics and Analytics , Quality and Process Standards and Guidelines , ISO 9000 , SWEBOK , ISO 15504.	05	10
8	DEVOPS: Overview, Problem Case Definition, Benefits of Fixing Application Development Challenges, DevOps Adoption Approach through Assessment, Solution Dimensions, What is DevOps? DevOps Importance and Benefits, DevOps Principles and Practices, 7 C's of DevOps Lifecycle for Business Agility, DevOps and Continuous Testing, How to Choose Right DevOps Tools, Challenges with DevOps Implementation, Must Do Things for DevOps, Mapping My App to DevOps -Assessment, Definition, Implementation, Measure and Feedback.	04	08
9	DIFFERENT TOOLS USED IN SE: UI and UX and related tools-Figma. Developer tool used for Code Review: SonarQube. To create Test cases: JUnit or NUnit. QA performance Testing tools: Win runner and Load runner, Tools for Test Management and Bug Tracking.	04	08

Course Outcomes:

Sr. No.	CO Statement	Unit
CO-1	Create SRS (Software Requirement Specification) document and SPMP (Software Project Management Plan) document	3,5
CO-2	Apply the concept of Functional Oriented and Object-Oriented Approach for Software Design.	1
CO-3	Recognize how to ensure the quality of software products, different quality standards and software review techniques.	7
CO-4	Apply various testing techniques and test plans for softwares.	6
CO-5	Identify modern Agile Development , Devops Concept of Industry and emerging trends of software engineering.	2,8,9
CO-6	Able to create UML designs of different components of software.	4

Teaching & Learning Methodology:

The various methods or tools to teach the above subject:

1. Problem - based learning
2. Design Thinking
3. Cooperative-based learning
4. Competency-based learning

List of Experiments:

Total Hours: 28

Note: Either Individual or in a team perform following, In case of Team Make sure the selected application has enough modules as per team size. Define roles and responsibilities of each team member.

Sr. No.	Practical Name
1	Take a sample application.
2	Select any SE Model with respect to your application and justify it.
3	Prepare Software Requirement Specification Document.
4	Prepare Project Gantt Chart and Work-Breakdown Structure.
5	Prepare UML diagrams.
6	Prepare UI and UX
7	Identify possible risks and steps to manage them.
8	Prepare different test plans.
9	Execute test cases.
10	Prepare reports and metrics like Defect Density, Bug Summary Report.

Books Recommended:

1. Roger S. Pressman, "Software engineering- A practitioner's Approach", McGraw -Hill International Editions
2. Ian Sommerville, "Software engineering", Pearson educationAsia
3. Pankaj Jalote, "Software Engineering" – A Precise Approach Wiley
4. Ali Behhforoz & Frederick Hudson OXFORD, "Software Engineering Fundamentals".
5. Rajib Mall, "Fundamentals of software Engineering", Prentice Hall of India.

6. Deepak Gaikwad, Viral Thakkar, “DevOps Tools from Practitioner’s ViewPoint”, Wiley
7. Armando Fox and David Patterson ,”Engineering Software as a Service An Agile Software Approach”.
8. John M Nicolas, “Project Management for Business”, Engineering and Technology, Elsevier

List of Open Source Software/learning website:

1. <https://devops.com/most-popular-open-source-devops-tools/>
2. www.onesmartclick.com/engineering/software-engineering.html
3. www.sei.cmu.edu

CO-PO-PSO Matrix:

Co. No.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO-1	1	1	2		3	1			3	3	3	2	3	1
CO-2	2	2	2	1	2				1		1	1	1	2
CO-3	1	1		2	2	1			1		2	2	1	1
CO-4	2	3	2	2	3				1		1	2	1	1
CO-5	2				3							1	1	2
CO-6	2		1		3					1			2	1