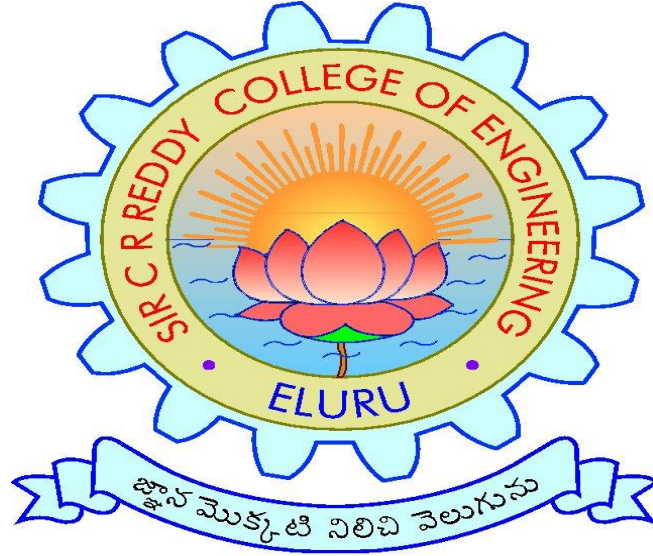**SIR C.R.REDDY COLLEGE OF ENGINEERING, ELURU**

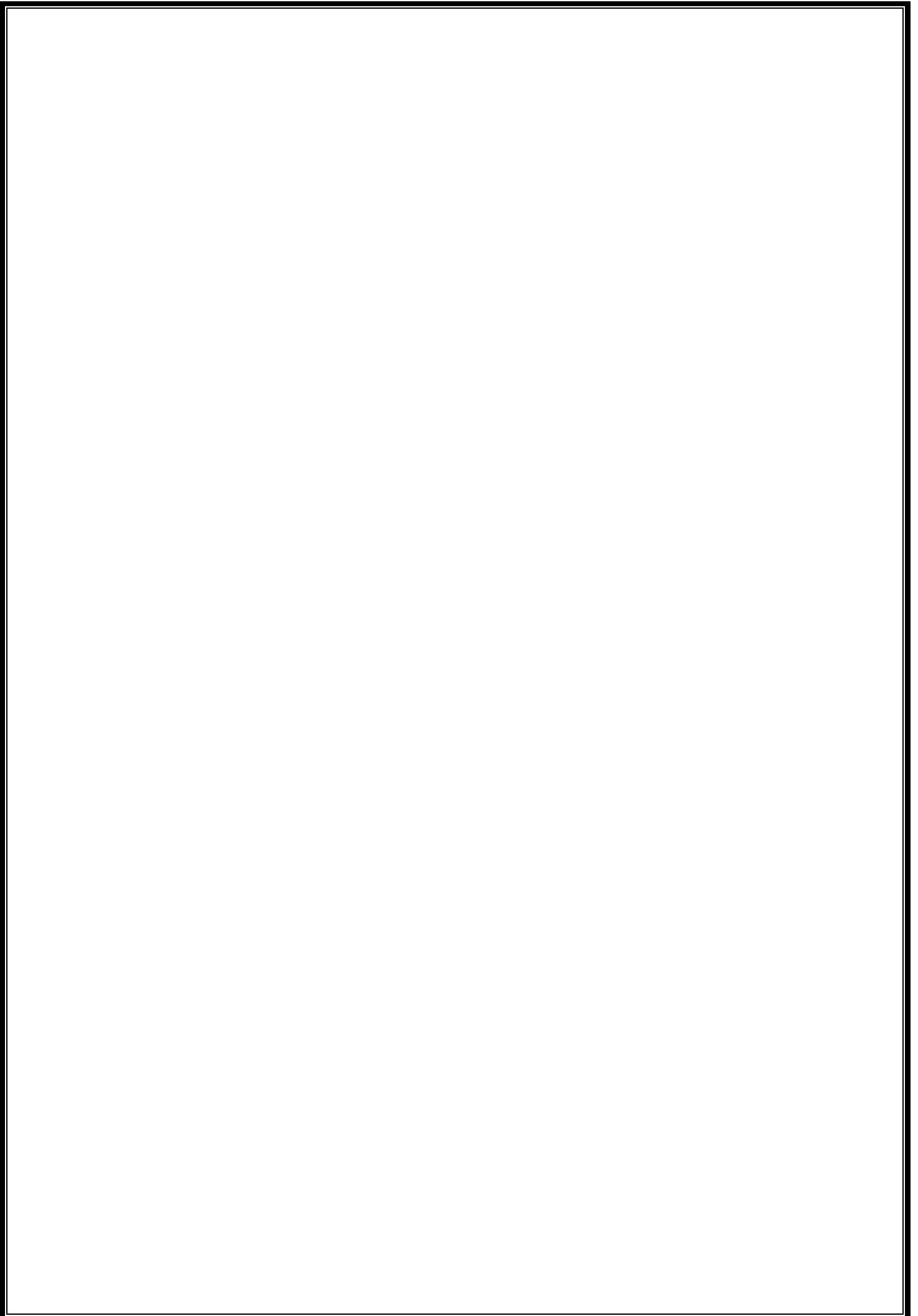**DEPARTMENT OF INFORMATION TECHNOLOGY**

COURSE HANDOUT

SUBJECT: PRINCIPLES OF SOFTWARE ENGINEERING (R2022122)

CLASS: III/IV B.Tech.

II SEMESTER A.Y. 2022-23

INSTRUCTORS : Dr. G.CHAMUNDESWARI (A & B SECS)

Course Handout Index

COLLEGE VISION

To emerge as a premier institution in the field of technical education and research in the state and as a home for holistic development of the students and contribute to the advancement of society and the region.

COLLEGE MISSION

To provide high quality technical education through a creative balance of academic and industry oriented learning; to create an inspiring environment of scholarship and research; to instill high levels of academic and professional discipline; and to establish standards that inculcate ethical and moral values that contribute to growth in career and development of society in general.

VISION OF THE DEPARTMENT

To be a premier Department in the region in the field of Information Technology through academic excellence and research that enable graduates to meet the challenges of industry and society

MISSION OF THE DEPARTMENT

- ❖ To Provide dynamic teaching-learning environment to make the students industry ready and advancement in career;
- ❖ To inculcate professional and leadership quality for better employability and entrepreneurship;
- ❖ To make high quality professional with moral and ethical values suitable for industry and society

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Solve real world problems through effective professional skills in Information Technology industry and academic research.

PEO2: Analyze and develop applications in Information Technology domain and adapt to changing technology trends with continuous learning.

PEO3: Practice the profession in society with ethical and moral values.

PROGRAM OUTCOMES (POs)

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using the first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/Development of Solutions: Design solutions for complex engineering problems and system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, society, and environmental considerations.

PO4: Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and Sustainability: Understand the impact of the professional engineering solutions in society and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and Team Work: Function effectively as an individual, and as a member or leader in diverse teams, and in multi-disciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multi-disciplinary environments.

PO12: Life-long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Design and develop software in the area of relevance under realistic constraints.

PSO2: Adopt new and fast emerging technologies in the field of Information Technology.

Directorate of Academic Planning
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA-533003, Andhra Pradesh, INDIA
(Established by AP Government Act No. 30 of 2008)

Lr. No. DAP/RAC/II Year /B. Tech/2022   *Date 02.11.2022*

Dr. KVSG Murali Krishna,
     M.E, Ph.D.,
Director, Academics & Planning
JNTUK, Kakinada

To
All the Principals of Affiliated Colleges,
JNTUK, Kakinada.

Revised Academic Calendar for II Year - B. Tech for the AY 2022-23
(2021-22 Admitted Batch)

I SEMESTER			
Description	**From**	**To**	**Weeks**
Community Service Project	**22.08.2022**	03.09.2022	2W
I Unit of Instruction	05.09.2022	29.10.2022	8W
I Mid Examinations	24.10.2022	29.10.2022	
II Unit of Instructions	31.10.2022	24.12.2022	8W
II Mid Examinations	19.12.2022	24.12.2022	
Community Service Project for Lateral Entry Students, Preparation & Practicals	26.12.2022	14.01.2022	3W
End Examinations	18.01.2023	28.01.2023	2W
Commencement of II Semester Class Work	**28.01.2023**		
II SEMESTER			
I Unit of Instructions	**30.01.2023**	25.03.2023	8W
I Mid Examinations	20.03.2023	25.03.2023	
II Unit of Instructions	27.03.2023	20.05.2023	8W
II Mid Examinations	15.05.2023	20.05.2023	
Preparation & Practicals	22.05.2023	27.05.2023	1W
End Examinations	29.05.2023	10.06.2023	2W

* As per the APSCHE Guidelines Out of the Total 180 hours of Community Service Project leading to 4 Credits, two weeks will be offline and remaining project work can be done during the II-I semester weekends and holidays.

Director,
Academics & Planning, JNTUK

Director
Academic Planning
JNTUK Kakinada

Copy to the Secretary to the Hon'ble Vice Chancellor, JNTUK
Copy to Rector, JNTUK
Copy to Registrar, JNTUK
Copy to Director Academic Audit, JNTUK
Copy to Director of Evaluation, JNTUK

Department Academic Calendar

Sir C R Reddy College of Engineering :: Department of Information Technology
II/IV B.Tech Academic Calendar for 2022-23

2022-23	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M
Jul 22					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
	27	28	29	30	31																									
Aug 22		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
	30	31																												
Sep 22				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
	28	29	30																											
Oct 22							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
	25	26	27	28	29	30	31																							
Nov 22		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
	30																													
Dec 22				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
	28	29	30	31																										
Jan 23	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
	31																													
Feb 23				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
	28																													
Mar 23				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
	28	29	30	31																										
Apr 23							1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
	25	26	27	28	29	30																								
May 23		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
	30	31																												
Jun 23					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
	27	28	29	30																										

List of Holidays			Mid exams
July 10: Bakrid	Oct 5: Vijayadasami	Mar 8 : holi	End Examinations
Aug 9:Moharum	Oct 9: Maulud Nabi	Mar 22 : Ugadhi	Commencement of Class work
Aug 15:Independence day	Oct 24 :Diwali	Mar 30 :Srirama navami	Workshops
Aug 19:Janmastami	Dec 25 :Christmas	Apr 5: Babu Jagjivan Ram Jayanti	Department fest/Elite
Aug 31:Ganesh Chaturdi	Jan 14-16: sankranti	Apr 7: Good friday	Community Service Project
Oct 2: Gandhi jayanthi	Jan 26:Republic Day	Apr 14:Ambetkar Jayanthi	HoD
	Feb 18 :Sivaratri	Jun 29: Bakrid	Department of IT

Course Description

Principles of Software Engineering focus on empowerment to handle the complexity of the design process in a more effective and efficient manner. These principles help reduce the effort required for designing software systems and even lower the chances of you introducing errors during the design process.

Course Objectives

The course is designed to :

- Give exposure to phases of Software Development, common process models including Waterfall, and the Unified Process, and hands-on experience with elements of the agile process.
- Give exposure to variety of Software Engineering practices such as requirements analysis and specification, code analysis, code debugging, testing traceability, and version control.
- Give exposure to Software Design techniques.

Course Outcomes

Students are able to

CO No's	COs	Level
CO1	Understand the basic concepts of software engineering, phases of software development in generic process models, unified and agile process models.	L2
CO2	Apply various engineering practices such as requirements analysis and specification, modeling, code analysis, testing, and quality assurance strategies to develop software.	L3
CO3	Analyze the gathered requirements to create various requirement models.	L4
C04	Prepare the architectural design, components level design, interface design and acquire skills to design and implement test cases at the Unit and Integration level.	L5

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	-	-		-	-	-	-	-	-	-	-	1	1
CO3	-	3	-	-	-	-	-	-	-	-	-	-	1	1
CO4	-	-	2	-	-	-	-	-	1	1	-	-	1	1
Avg	3	3	2	-	-	-	-	-	1	1	-	-	1	1

Lesson Plan

S.No	Unit	Description	Teaching Aids	CO
1.		Introduction	BB	CO1
2.		The Nature of Software, The Unique Nature of WebApps	BB	CO1
3.		Software Engineering, The Software Process	BB	CO1
4.	I	Software Engineering Practice, Software Myths, How It All Starts	BB	CO1
5.		A Generic Process Model, Process Assessment and Improvement	BB	CO1
6.		Prescriptive Process Models, Specialized Process Models	PPT	CO1
7.		The Unified Process, Personal and Team Process Models, Process Technology.	BB	CO1
8.		Agility, Agility and the Cost of Change, Agile Process, Extreme Programming (XP)	BB	CO1

9

9.	**II**	Other Agile Process Models, A Tool Set for the Agile Process	PPT	CO2
10.		A Tool Set for the Agile Process	BB	CO2
11.		Software Engineering Knowledge	BB/PPT	CO1
12.		Core Principles, Principles That Guide Each Framework Activity	BB/PPT	CO2
13.		Requirements Engineering, Establishing the Groundwork	BB/PPT	CO2
14.		Eliciting Requirements, Developing Use Cases	BB	CO3
15.		Building the Requirements Model	PPT	CO3
16.		Negotiating Requirements	BB	CO3
17.		Validating Requirements	BB	CO3
18.	**III**	Requirements Analysis	BB	CO3
19.		Scenario-Based Modeling	BB/PPT	CO2
20.		UML Models That Supplement the Use Case	BB/PPT	CO1
21.		Data Modeling Concepts	BB/PPT	CO2
22.		Class-Based Modeling	PPT	CO1
23.		Requirements Modeling Strategies	BB	CO1
24.		Flow-Oriented Modeling	BB	CO2
25.		Creating a Behavioral Model	PPT	CO4
26.		Patterns for Requirements Modeling	BB	CO3
27.		Requirements Modeling for WebApps	BB	CO2
28.	**IV**	Design within the Context of Software Engineering	BB	CO3
29.		The Design Process	BB/PPT	CO1
30.		Design Concepts	BB	CO1
31.		The Design Model	BB/PPT	CO1
32.		Software Architecture	BB/PPT	CO2
33.		Architectural Genres	BB	CO1
34.		Architectural Styles	BB/PPT	CO2
35.		Assessing Alternative Architectural Designs	BB/PPT	CO3
36.		Architectural Mapping Using Data Flow	BB	CO4
37.		What Is a Component?	BB	CO1
38.		Designing Class-Based Components	BB/PPT	CO4
39.		Conducting Component-Level Design	BB	CO4
40.		Component-Level Design for WebApps	BB	CO4

41.		Designing Traditional Components	BB	CO4
42.		Component-Based Development	BB	CO4
43.		The Golden Rules, User Interface Analysis and Design	BB	CO2
44.		Interface Analysis, Interface Design Steps	BB	CO3
45.		WebApp Interface Design	BB/PPT	CO4
46.		Design Evaluation, Elements of Software Qualtiy Assurance	BB/PPT	CO4
47.		SQA Tasks, Goals & Metrics	BB/PPT	CO2
48.		Statistical SQA, Software Reliability	BB	CO3
49.	V	A Strategic Approach to Software Testing, Strategic Issues	BB/PPT	CO2
50.		Test Strategies for Conventional Software	BB/PPT	CO2
51.		Test Strategies for Object-Oriented Software	BB/PPT	CO2
52.		Test Strategies for WebApps	BB/PPT	CO2
53.		Validation Testing, System Testing	BB/PPT	CO2
54.		The Art of Debugging, Software Testing Fundamentals	BB/PPT	CO1
55.		Internal and External Views of Testing	BB/PPT	CO2
56.		White-Box Testing, Basis Path Testing	BB	CO2
Total Classes			**56**	

Evaluation Pattern

S. No	Components	Internal	External	Total
1	Theory	30	70	100
2	Engineering Graphics/Design/Drawing	30	70	100
3	Practical	15	35	50
4	Mini Project/Internship/Industrial Training/ Skill Development programmes/Research Project	–	50	50
5	Project Work	60	140	200

Marks Range Theory (Max – 100)	Marks Range Lab (Max – 50)	Level	Letter Grade	Grade Point
≥ 90	≥ 45	Outstanding	A+	10
≥80 to <89	≥40 to <44	Excellent	A	9
≥70 to <79	≥35 to <39	Very Good	B	8
≥60 to <69	≥30 to <34	Good	C	7
≥50 to <59	≥25 to <29	Fair	D	6
≥40 to <49	≥20 to <24	Satisfactory	E	5
<40	<20	Fail	F	0
-		Absent	AB	0

TIME TABLE

	1	2	3	4		5	6	7	8
DAY	9-9.50	9.50-10.40	11-11.50	11.50-12.40		1.40-2.30	2.30-3.20	3.20-4.10	4.10-5.00
MON	PSE(B)			PSE(A)(T)		UML LAB(A)			
TUE		PSE(A)		PSE(B)					
WED			PSE(B)						
THU		UML LAB(B)				PSE(B)			
FRI	PSE(A)					PSE(A)			
SAT		PSE(B)(T)		PSE(A)					

SYLLABUS

UNIT-I
The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths. A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology.

UNIT-II
Agility, Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, A Tool Set for the Agile Process, Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity, Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

UNIT-III
Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling, Requirements Modeling Strategies, Flow-Oriented

Modeling, Creating a Behavioral Model, Patterns for Requirements Modelling, Requirements Modeling for WebApps.

UNIT-IV
Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model, Software Architecture, Architectural Genres, Architectural Styles,Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow, Components, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component- Based Development.

UNIT-V
The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation, Elements of Software Qualtiy Assurance, SQA Tasks, Goals & Metrics, Statistical SQA, Software Reliability, A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing.

TEXT BOOKS

1. Software Engineering a practitioner's approach, Roger S. Pressman, Seventh Edition, McGraw Hill Higher Education.

2. Software Engineering, Ian Sommerville, Ninth Edition, Pearson.

REFERENCE BOOKS

1. Software Engineering, A Precise Approach, PankajJalote, Wiley India, 2010.

2. Software Engineering, Ugrasen Suman, Cengage.

UNIT WISE Important questions

UNIT - 1
LONG ANSWER QUESTIONS

1. Explain the unique nature of web applications?
2. Define software engineering and explain the layers of software engineering?
3. Define a software process and explain the generic software process framework?
4. What are the Umbrella activities of a software process?
5. Explain the essence of software engineering practice?
6. What is a software myth and explain various types of myths?

7. Explain about process assessment and improvement?
8. Explain software development life cycle. Discuss various activities during SDLC.
9. Explain the following software process models?
    i.   Waterfall model
   ii.   V model
  iii.   Incremental model
   iv.   Evolutionary models: prototyping model, spiral model, concurrent model
    v.   Other models
            a. Unified process approach
            b. Specialized process model
            c. Personal software process
            d. Team software process

UNIT - 2
LONG ANSWER QUESTIONS

1. List agility principles for those who want to achieve agility
2. Explain other Agile Process Models in briefly?
3. Explain Extreme Programming Process Model?
4. Explain the core principles that guide each framework activity?
5. List out the seven distinct tasks of requirements engineering?
6. Explain the elements of building requirements model?
7. Explain the process of Eliciting requirements of a project?
8. Explain the concept of negotiating and validating requirements?

UNIT - 3
LONG ANSWER QUESTIONS

1. Describe the process of Scenario based modeling.
2. Explain data modeling concepts with an example.
3. Explain about class based modeling concepts with an example.
4. Explain flow oriented modeling concepts with an example.
5. Explain behavior modeling concepts with an example.

6. What is use case diagram? Draw a use case diagram for an safe home surveillance system?
7. Explain activity diagrams? Draw an activity diagram for an safe home surveillance system?
8. Explain swim lane diagrams? Draw a swim lane diagram for an safe home surveillance system?
9. What is class diagram? Draw class diagram for an safe home surveillance system?
10. What is sequence diagram? Draw a sequence diagram for any system?
11. Illustrate state chart diagrams? Draw a state chart diagram for any system?
12. Illustrate data flow diagrams with the help of examples?
13. What is ER diagram? Explain with the help of examples?
14. Explain Patterns for Requirements Modeling?
15. Explain Requirements Modeling for WebApps?

UNIT – 4
LONG ANSWER QUESTIONS

1. Map the components of Analysis model with the respective components of Software design Model.
2. Explain the software design process in detail
3. Explain each of the following with respect to "software design"?
   a)Abstraction b) Architecture c) Patterns d) Separation of concerns
   e)Modularity   f) Hiding g) Functional independence h) Refinement
   i)Aspects j) Refactoring k) OO design concepts l) DesignClasses
4. What is software architecture? Expand the role of software architectural design.
5. What are the different architectures required in architectural styles.
6. Explain the steps required for architectural mapping using data flow.
7. Explain object oriented view and traditional view in a component level design.
8. Design a class based component with an example.
9. What are the steps required for conducting component level design.
10. Explain the concepts of designing class based components?
11. Explain about component level design for Web Apps?
12. Explain about component based software engineering?

UNIT – 5
LONG ANSWER QUESTIONS

1. State and explain Golden rules of User Interface
2. Write a short notes on User Interface Analysis and design.
3. Explain user interface analysis with example
4. Write a short notes on user interface design steps
5. What are the steps required for interface design workflows for webapps.
6. Write a short notes on design evaluation.
1. What are the steps required for strategic approach in Testing
2. What is the objective of unit testing? Explain
3. Explain integration testing in detail?
4. Explain strategic issues in Testing?
5. Write a short notes on validation Testing?
6. Explain System testing in detail
7. Explain the art of debugging?
8. What are the characteristics that lead to testable software?
9. Explain basis path testing in detail with an example?
10. Explain about graph matrix?