SETHU INSTITUTE OF TECHNOLOGY

(An Autonomous Institution | Accredited by NAAC with 'A++' Grade)
PULLOOR, KARIAPATTI – 626 115.

DEPARTMENT OF INFORMATION TECHNOLOGY

Institute Vision	To promote excellence in technical education and scientific research for the benefit of the society
Institute Mission	• To provide quality technical education to fulfill the aspiration of the student and to meet the needs of the Industry • To provide holistic learning ambience • To impart skills leading to employability and entrepreneurship • To establish effective linkage with industries • To promote Research and Development activities • To offer services for the development of society through education and technology **Core Values** • Quality • Commitment • Innovation • Team work • Courtesy **QUALITY POLICY** • To provide Quality technical education to the students • To produce competent professionals and contributing citizens • To contribute for the upliftment of the society

INSTRUCTIONAL SYSTEM DESIGN

Course Code	**21UIT405**		
Course Name	**Software Engineering Methodologies**		
Name of the Course Coordinator	**Dr. C.Parameswari ASP/IT**		
Name of the Course Instructor	**Dr.C.Parameswari ASP/IT** **Mr.P.Ashok Ram AP/IT**		
Academic Year	**2023-2024 EVEN**		
Branch & Class *(Year/Sem)*	**IT & II YEAR / IV SEM**		
Effective Date	**18.12.2023**	**Revision No.:**	**0**

Prepared By,
Dr.C.Parameswari ASP/IT
Mr.P.Ashok Ram AP/IT

Approved By,
HOD/IT

DEPARTMENT VISION

- To promote excellence in producing competent IT professionals to serve the society through technology and research.

DEPARTMENT MISSION

- Producing Competent Professionals in Information and Communication Technologies
- Educating the Students with the State of Art Computing Environment and Pedagogical Innovations
- Encouraging Entrepreneurship and Imparting Skills for Employability
- Establishing Collaboration with IT and Allied Industries
- Promoting Research in Information and Communication Technology to Improve the Quality of Human Life
- Offering Beneficial Service to the Society by Inculcating Knowledge and Providing IT Solutions

PROGRAM EDUCATIONAL OBJECTIVES

PEO 1	Exhibit Proficiency in Analyzing, Designing and Developing IT Based Solutions to Cater to the Business and Societal Needs. **{Technical Competence}**
PEO 2	Provide Professional Expertise to the Industry and Society with Effective Communication and Ethics. **{Professionalism}**
PEO 3	Engage in Lifelong Learning for Professional Development and Research. **{Life-Long Learning}**

PROGRAM SPECIFIC OUTCOMES	
PSO – 1	Design Software Solutions Using Programming Skills and Computing Technologies.
PSO – 2	Design and Implement Data Communication System Using Various IT Components.

PROGRAM OUTCOMES	
1.	Apply the knowledge of Mathematics, Basic Science, Computer and communication Fundamentals to solve complex problems in Information Technology. **[Engineering Knowledge]**
2.	Identify, formulate, review research literature and analyze complex problems reaching concrete conclusions using principles of mathematics, Engineering sciences and Information Technology. **[Problem Analysis]**
3.	Design solution for complex information and communication engineering problems and design system components or processes that meet with realistic constraints for public health and safety, cultural, societal and environment considerations. **[Design/Development of Solutions]**
4.	Conduct investigations of complex Information technology related problems using research based knowledge and research methods including design of experiments, analysis and interpretation of data to provide valid conclusions through synthesis of information. **[Conduct investigations of complex problems]**
5.	Create, select and apply appropriate techniques, resources and modern IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations. **[Modern Tool Usage]**

6.	Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and consequent responsibilities relevant to professional engineering practice. **[The Engineer and Society]**
7.	Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development. **[Environment and sustainability]**
8.	Apply ethical principles and commit to professional ethics and responsibilities through the norms of professional engineering practice. **[Ethics]**
9.	Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings. **[Individual and Team Work]**
10.	Communicate effectively with the engineering community and the 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. **[Communication]**
11.	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member /or leader in a team, to manage projects in multi-disciplinary environment. **[Project Management and Finance]**
12.	Recognize the need for, and have the preparation and ability to engage in independent and Life-long learning in broadest context of technological change. **[Life-long Learning]**

Course Code	Course Name	L	T	P	C
21UIT405	Software Engineering Methodologies (Common to IT and CSE)	**3**	**0**	**0**	**3**

Course Objectives:

- Understand the phases in a software project
- Understand fundamental concepts of requirements engineering and Analysis Modeling
- Understand the major considerations for enterprise integration and deployment
- Learn various testing and maintenance measures

Unit I Software Process	9
Introduction to Software Engineering, Software Process, Prescriptive Process Models and Specialized Process Models - Agile Process Model - Agile Manifesto and Principles	
Unit II Requirement Analysis and Specification	9
Software Requirements: Functional and Non-Functional, User requirements, System Requirements, Software Requirements Document - Requirement Engineering Process: Feasibility Studies, Requirements Elicitation and Analysis, Requirements Validation, Requirements Management-Classical Analysis: Structured System Analysis, Petri Nets -Data Dictionary	
Unit III Software Design and Development	9
Design Process – Design Concepts – Design Model-Design Heuristic – Architectural Design – Architectural Styles, Architectural Design, Architectural Mapping using Data Flow – User Interface Design: Interface Analysis, Interface Design – Introduction to Real Time Software Design – Component Level Design: Designing Class Based Components, Traditional Components	
Unit IV Software Testing and Maintenance	9
Software Testing Fundamentals – Internal and External Views of Testing- White Box Testing - Basis Path Testing - Control Structure Testing - Black Box Testing - Regression Testing - Unit Testing - Integration Testing - Validation Testing - System Testing and Debugging - Software Implementation Techniques: Coding Practices - Refactoring-Maintenance and Reengineering - BPRModel-Reengineering Process Model - Reverse and Forward Engineering.	
Unit V Project Management	9
Software Project Management: Estimation, Make / Buy Decision,COCOMO-II-Project Planning - Project Scheduling - Risk Management - RMMM Plan - CASE Tools	
Total: 45 Periods	

Text Books:

1. Roger Pressman.S, "Software Engineering – A Practitioner's Approach", McGraw Hill International Edition, 9th Edition, 2019.
2. Ian Sommerville, "Software Engineering", Pearson Education Asia 10th Edition, 2017

References:

1. RajibMall, "Fundamentals of Software Engineering", PHILearning Private Limited, 3rd Edition, 2009.
2. Pankaj Jalote, "Software Engineering, A Precise Approach Fundamentals of Software Engineering", Wiley India, 2010.
3. Stephen R.Schach, "Software Engineering", Tata McGraw-Hill Publishing Company Limited, 2007.

Course Outcomes

At the end of the course the student will be able to

COs	Course Outcomes Statement	Taxonomy	Domain	PO Mapping
CO1	Elaborate the software engineering methodologies and project management techniques	Understand	Cognitive	-
CO2	Apply the software development and management techniques for real time projects in agile scenario	Apply	Cognitive	PO1
CO3	Analyze various software methodologies, design techniques, testing strategies by means of software project management	Analysis	Cognitive	PO2
CO4	Design a process model which suits the business need	Create	Cognitive	PO3
CO5	Use modern tools to demonstrate the software engineering process	Apply	Cognitive	PO5
CO6	Work individually and as a member in multidisciplinary teams	Value	Affective	PO9

Continuous Internal Assessment Apportionment (for 40 Marks of CIE - Theory)

Assessment	Component	Assessment Details	Marks	Convert
I	Internals	Periodical Test - I	12	60
	Assignment	Group	80	40
II	Internals	Periodical Test - II	12	60
	Assignment	Individual	40	40
	Total			40

Lecture Schedule 21UIT405 – Software Engineering Methodology				
Lecture#	**Unit#**	**Topic**	**Text Book**	**Mode of Delivery**
Unit I Software Process				
1	I	Introduction to Software Engineering	T2 (3)	Lecture
1	I	Software Process	T2 (27)	Lecture
1	I	Prescriptive Process Models	T2 (29)	Lecture
1	I	PPM- Evolutionary Process Model	T2 (35)	Lecture
2	I	PPM- Concurrent Evolutionary Process	T2 (41)	Lecture
1	I	Specialized Process Models	T2 (46)	Lecture
1	I	Agile Process Model	T2 (56)	Lecture
1	I	Agile Manifesto and Principles	T2 (72)	Lecture
Unit II Requirement Analysis and Specification				
1	II	Software Requirements: Functional and Non-Functional	T2 (84)	Lecture
1	II	User requirements, System Requirements	T2 (87)	Lecture
1	II	Software Requirements Document	T2 (91)	Lecture
1	II	Requirement Engineering Process: Feasibility Studies	T2 (99)	Lecture
1	II	Requirements Elicitation and Analysis	T2 (100)	Lecture
1	II	Requirements Validation	T2 (110)	Lecture

1	II	Requirements Management	T2 (111)	Lecture
1	II	Classical Analysis: Structured System Analysis	www	Lecture
1	II	Petri Nets -Data Dictionary	www	Lecture
Unit III Software Design and Development				
1	III	Design Process – Design Concepts	T1(338)	Lecture
1	III	Design Model-Design Heuristic	T1(355)	Lecture
1	III	Architectural Design – Architectural Styles	T1(365)	Lecture
1	III	Architectural Mapping using Data Flow	T1(378)	Lecture
1	III	Interface Analysis, Interface Design	T1(401)	Lecture
1	III	Introduction to Real Time Software Design	T1(465-471)	Lecture
1	III	Component Level Design	T2 (423)	Lecture
1	III	Designing Class Based Components	T2 (430)	Lecture
1	III	Traditional Components	T2 (435)	Lecture
Unit IV Software Testing and Maintenance				
1	IV	Software Testing Fundamentals – Internal and External Views of Testing	T2 (438)	Lecture
1	IV	White Box Testing - Basis Path Testing - Control Structure Testing	T2 (444)	Lecture
1	IV	Black Box Testing - Regression Testing	T2 (459)	Lecture
1	IV	Unit Testing - Integration Testing	T2 (485)	Lecture
1	IV	Validation Testing - System Testing and Debugging	T2 (495)	Lecture
1	IV	Software Implementation Techniques: Coding Practices	T2 (503)	Lecture
1	IV	Refactoring	T2 (508)	Lecture
1	IV	Maintenance and Reengineering - BPRModel	T2 (802)	Lecture
1	IV	Reengineering Process Model - Reverse and Forward Engineering.	T2 (809)	Lecture
Unit V Project Management				
2	V	Software Project Management: Estimation	T2 (123)	Lecture
1	V	Make / Buy Decision	T2 (130)	Lecture

2	V	COCOMO-II	T2 (133)	Lecture
1	V	Project Planning	T2 (113)	Lecture
1	V	Project Scheduling	T2 (165)	Lecture
1	V	Risk Management	T2 (145)	Lecture
1	V	RMMM Plan	T2 (159)	Lecture
Augmentation Topic: Product Burn down chart			WWW	

Micro-level Audit of Assessment Tool(s) used

Test Name	Q. No.	Understand	Apply	Analyze	Create	Design (Modern Tool)	Any other (Team Work)	Total Marks
Test - I	1	2						2
	2	2						2
	3	2						2
	4	2						2
	5	2						2
	6	2						2
	7		2					2
	8		2					3
	9		2					2
	10	16						16
	11		16					16
Test - II	1	2						2
	2	2						2
	3	2						2
	4	2						2
	5	2						2
	6	2						2
	7		2					2
	8	2						2
	9	2						2
	10		16					16
	11	16						16
Assignment	I			20	20	20	20	80
Assignment	II			10	10	10	10	40
Total Marks		**60**	**40**	**30**	**30**	**30**	**30**	**220**

QUESTION PATTERN (PERIODICAL TEST)

9*2 marks =18 marks

2*16marks=32 marks

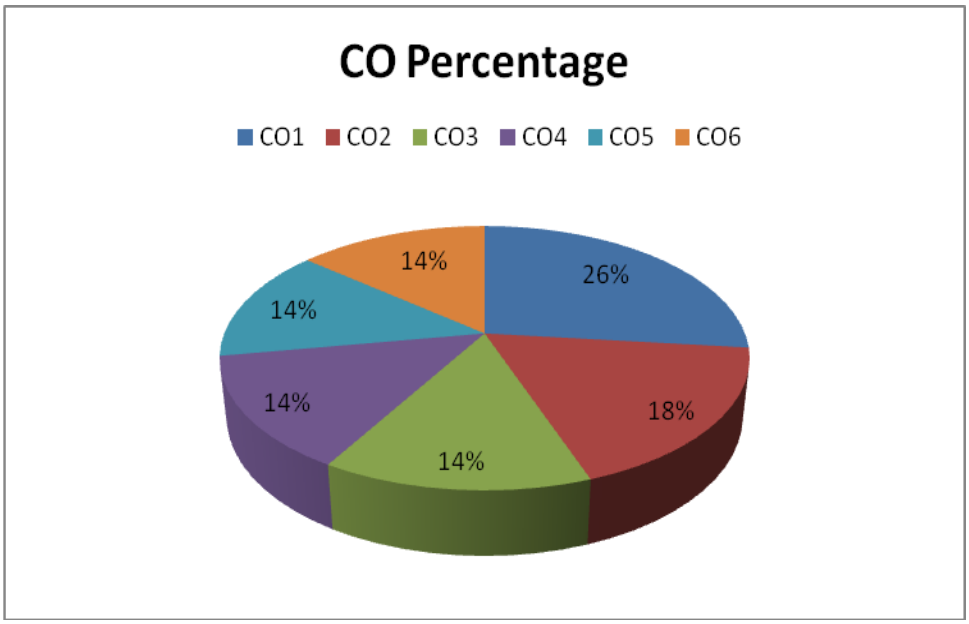
QUESTION PATTERN (END SEMESTER EXAM)

10*2 marks =20 marks

5*16marks=80 marks

Assessment Tool Mapping to CO - Theory

Test Name	CO1	CO2	CO3	CO4	CO5	CO6	Total
Test–I	28	22					50
Test-II	32	18					50
Assignment I			20	20	20	20	80
Assignment II			10	10	10	10	40
Total Marks	**60**	**40**	**30**	**30**	**30**	**30**	**220**
Percentage	**27**	**18**	**14**	**14**	**14**	**14**	**100**



CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1														
CO2	**3**												**2**	
CO3		**3**											**2**	
CO4			**3**										**2**	
CO5					**3**								**2**	
CO6									**3**				**2**	

S. No.	Percentage Questions towards GA attribute (50 Marks)	Level (Weight)
1.	**>=12% of total marks(50)**	**3**
2.	**>=6% and <=11 of total marks(50)**	**2**
3.	**<=5% of total marks(50)**	**1**

Group Assignment – I

Create Software Requirement Specification (SRS), Use Case diagram; apply any one of the perspective process model.

The Rubrics used for evaluation of the assignment - I is shown in the following table.

S. No	Criteria	Outstanding (16-20)	Very Good (12-15)	Good (11-14)	Satisfactory (6-10)	Unsatisfactory (1-5)
1.	Requirement Analysis (CO3)	Completeness of requirement specified	Obtain the number of unique requirements	Understandable requirements	Poor verifiability of the requirement	Difficult to trace the requirement
2.	Business Process Modeling Notation (CO4)	Clearly define the scope of the process	Visually depicts a detailed sequence of business activities	Communicate the process flow with both non-technical and technical	Improper Notation representation	Modeling organizational structures and data flow models are not appropriate
3.	SRS Documentation (CO5)	Correctness of requirements stated in the SRS	Covering all the functional and non-functional requirements properly	No conflicts between any set of requirements	Text containing information irrelevant to the problem	Same thing described at several places in different ways
4.	Team Work (CO6)	All four members contributed equally well.	Two team members did not contribute equally well.	Three team members did not contribute equally well.	Contribution of each member was not adequate	Contribution of each member was poor

Assignment – II

Create a video on how to use the Junit Tool for testing. Demonstrate a testing case with the tool and post the video in your YouTube channel. Receive comments from your peers and improve your video accordingly. 40 Marks

Scheme of Evaluation

Sl.no	Contents	Mark Allocation
1	**Writing suitable test cases(CO3)**	**10**
2	**Testing with suitable coding(CO4)**	**10**
3	**Usage of Modern Tool(CO5)**	**10**
4	**Peer Review and corrections made**	**10**
Total		**40 marks**

Rubrics for level of performance

Writing suitable test cases(CO3)	Identification of all the test cases for the problem taken (9-10)	Partial identification (5-8)	Poor identification (1-4)
Testing with suitable coding(CO4)	Exact code with proper assertations (9-10)	Very few failed test cases (5-8)	Major false positives (1-4)
Usage of Modern Tool(CO5)	Proper implementation and demonstration of Junit (9-10)	Proper implementation but with limited demonstration (5-8)	Poor demonstration (1-4)
Peer Review and corrections made	Proper reply to the peer review comments (9-10)	No reply to the peers but corrections are made (5-8)	Reaction to comments, but poor correction carried out (1-4)

Competency addressed in the Course and Corresponding Performance Indicators

PO	Competency	Performance Indicators
PO1	1.6 Demonstrate competence in engineering fundamentals 1.7 Demonstrate competence in specialized engineering knowledge to the program	1.6.1 Apply engineering fundamentals 1.7.1 Apply theory and principles of computer science engineering to solve an engineering problem
PO2	2.1 Demonstrate an ability to identify and formulate complex engineering problem 2.6 Demonstrate an ability to formulate a solution plan and methodology for an engineering problem 2.8 Demonstrate an ability to execute a solution process and analyze results	2.5.1 Evaluate problem statements and identifies objectives 2.6.2 Identify functionalities and computing resources 2.6.3 Identify existing solutions/methods to solve the problem, including forming justified approximations and assumptions 2.6.5 Compare and contrast alternate solution processes to select the best process

		2.8.3 Identify the limitations of the solutions and sources/causes 2.8.4 Arrive at conclusions with respect to the objectives
PO3	3.5 Demonstrate an ability to define a complex open ended problem in engineering terms	3.5.1 Able to define a precise problem statement with objectives and scope 3.5.6 Able to develop software requirement specifications
	3.6 Demonstrate an ability to generate a diverse set of alternative design solutions	3.6.1 Able to explore design alternatives 3.6.3 Identify suitable non functional requirements for evaluation of alternate design solutions
	3.8 Demonstrate an ability to advance an engineering design to defined end state	3.8.1 Able to refine architecture design into a detailed design within the existing constraints 3.8.2 Able to implement and integrate the modules 3.8.3 Able to verify the functionalities and validate the design
PO5	5.4Demonstrate an ability to identify/create modern engineering tools, techniques and resources.	5.4.1 Identify modern engineering tools, techniques and resources for engineering activities
PO9	9.4 Demonstrate an ability to form a team and define a role for each member 9.6 Demonstrate success in a team based project	9.4.2 Implement the norms of practice of effective team work , to accomplish a goal 9.6.1 Present results as a team, with smooth integration of contributions from all individual efforts

Understand Level			
Unit 1			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Compare evolutionary model with waterfall model	2	Test 1
2.	List the categories of software	2	Test 1
3.	What is component based development?	2	Test 1

4.	Which process model is suitable for handling projects in Fastener manufactures?	2	Test 1
5.	What is the concept of agile model?	2	Test 1
6.	What are the merits of incremental model?	2	Test 1
7.	Name the evolutionary process models.	2	Test 1
8.	Explain in detail the umbrella activities in the aspect of maintaining the quality.	16	Test 1
9.	Differentiate the features of various software lifecycle models by highlighting their advantages and disadvantages.	16	Test 1
10.	Explain in detail the types of an agile framework.	16	Test 1
Unit 2			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	What is data dictionary?	2	Test 1
2.	What are the steps in structured system analysis?	2	Test 1
3.	Write the benefits of feasibility study.	2	Test 1
4.	Why do we use PetriNets?	2	Test 1
5.	Classify the following as functional or non-functional requirements for a banking system. a) Verifying bank balance b) Completion of transactions in less than one second c) Withdrawing money from bank d) Extending the system by providing more tellers for the customers.	2	Test 1
6.	What are the two types of system requirements?	2	Test 1
7.	What are the metrics for specifying nonfunctional requirements?	2	Test 1
8.	Differentiate PSPEC and CSPEC.	2	Test 1
9.	Briefly explain the requirement engineering process	16	Test 1
10.	Explain the process of software requirement specification with example	16	Test 1

Unit 3			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Differentiate refactoring and refinement	2	Test 2
2.	Differentiate cohesion and coupling with an example	2	Test 2
3.	Why do we need interface design?	2	Test 2
4.	What are the benefits of horizontal partitioning?	2	Test 2
5.	What are the various design patterns?	2	Test 2
6.	What are the Golden rules of UI?	2	Test 2
7.	What is meant by FURPS attributes with respect to design quality?	2	Test 2
8.	What are four phases of a software design model?	2	Test 2
9.	Discuss the steps involved in translating the data flow into software architecture.	16	Test 2
10.	Illustrate various architectural design concepts of software Engineering.	16	Test 2
Unit 4			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Differentiate verification and validation.	2	Test 2
2.	What is meant by integration testing?	2	Test 2
3.	What is black box testing?	2	Test 2
4.	Who will perform white box testing? Why?	2	Test 2
5.	What are the types of software maintenance?	2	Test 2
6.	What are the various black box testing techniques?	2	Test 2
7.	What are the reasons behind performing white box testing?	2	Test 2
8.	Explain the White box testing techniques with example	16	Test 2
9.	Explain the following i) Integration Testing ii) System Testing	16	Test 2

10.	Distinguish graph based testing, boundary value analysis, equivalence partitioning and comparison testing.	16	Test 2
Unit 5			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	What is COCOMO-II?	2	Test 2
2.	List some of the project planning techniques.	2	Test 2
3.	What is RMMM Plan?	2	Test 2
4.	What are the case tools?	2	Test 2
5.	List some of the project estimation techniques.	2	Test 2
6.	How the risks are mitigated?	2	Test 2
7.	Give an example for Make/Buy situation.	2	Test 2
8.	How will you plan and schedule a software project?	16	Test 2
9.	Explain COCOMO-II Model.	16	Test 2
10.	Explain the necessity of risk management in a project.	16	Test 2
11.	What are project metrics? How it is calculated?	16	Test 2

Apply Level			
Unit 1			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Assume that you are the technical manager of a software development organization. A client is approaching you for a software solution. The problems stated by the client have some uncertainties which lead to loss if it is not planned and solved. What software development model you will suggest for this project? Justify.	2	Test 1
2.	Draw a use case diagram for an online shopping which should provide provisions for registering, authenticating the customers and also for online payment through any of the payment system like paypal.	2	Test 1
3.	You have to develop a word processing software product, what process model will you choose? Justify your answer.	2	Test 1

4.	Suppose you must develop software for a client with minimum risk involved in development. But the client is not able to define the detailed input and output requirements. In this situation which software process model would you choose? Justify your answer and explain diagrammatic illustration.	16	Test 1
5.	An expert solution INC. is a company engaged in software consultancy and wants to set up specialized software. Development environment to support system software activities. There are no guidelines available with the Management for setting up such a center as they have so far only worked in application areas. Suggest a process model which can help the company set up its operations and develop software.	16	Test 1
6.	Which model needs to be chosen for a startup project? Explain your answer and process model.	16	Test 1
7.	Assume that you are the technical manager of a software development organization. A client approached you for a software solution. The problems stated by the client have uncertainties which lead to loss if it is not planned and solved. What software development model you will suggest for this project? Justify and Explain the process model in detail.	16	Test 1
Unit 2			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Draw the level 0 DFD for railway reservation system.	2	Test 1
2.	Draw an ER diagram for exam registration system.	2	Test 1
3.	Consider you are associated with a online shopping website designing project. List the functional and nonfunctional requirements for the website design with illustrations and justifications.	16	Test 1
4.	You are creating a website for Sethu Institute of Technology. Briefly explain the functional and nonfunctional requirements associated with the task.	16	Test 1
5.	Draw DFD level 0 and level 1for online hotel booking System.	16	Test 1
6.	Draw Petrinet for ATM Transaction.	16	Test 1
7.	Draw an ER Diagram for Student course registration system and Exam registration System in an university.	16	Test 1
8.	A petrol (gas) pump system that includes a credit card reader. The customer swipes the card through the reader and then specifies the amount of fuel required. The fuel is delivered and the customer's account debited. Write the functional and Nonfunctional requirements for the above case.	16	Test 1

Unit 3			
S.NO	**Questions**	**Marks**	**Assessment Tool**

1.	How many inputs are required for 100% decision coverage?
	Read A
	Read B
	IF A+B >100 THEN
	Print (A+B is Large)
	ENDIF
	If A >50 THEN
	Print (A is Large)
	ENDIF
		2	Test 2

2.	Illustrate liskov substitution principle with an example program	16	Test 2
3.	Draw the class diagram for Banking services and explain the relationship exists between the classes.	16	Test 2
Unit 4			
S.NO	**Questions**	**Marks**	**Assessment Tool**

1.	Given the following sample of pseudo code:
	Input number of Sandwich
	Input number of Iced tea
	If Sandwich > 0 and Iced tea > 0 then
	Input Do you want home delivery (Yes / No)
	If home delivery = No?
	Print Serve their order now
	End if
	End If
	Which of the following test cases will ensure that statement 6 is executed?
	a) Sandwich = 1, Iced tea = 1, home delivery = yes
	b) Sandwich = 1, Iced tea = 1, home delivery= no
		2	Test 2

	c) Sandwich= 1, Iced tea = 2, home delivery= yes d) Sandwich= 1, Iced tea = 0, home delivery = no		
2.	Give at least three examples in which black-box testing may give the impression that "everything's OK," while white-box tests may uncover an error. Give at least three examples in which white-box testing may give the impression that "everything's OK," while black-box tests may uncover an error.	16	Test 2
3.	Find the cyclomatic complexity for the below scenario. Find all the paths to be covered ① **start** ② **if** (X) **then** ③ **if** (Y) **then** ④ perform A perform B **else** ⑤ perform C perform D ⑥ **endif** ⑦ **endif** ⑧ **end**	16	Test 2
4.	Write test cases for the following constraints i) The length of Password should be 6-10 ii) Password should contain a special character	16	Test 2

5.	Refactor the below codes
	
	i)while True:
	mass = int(input("Enter the mass value: "))
	if mass > 0:
	break
	while True:
	acceleration = int(input("Enter the acceleration: "))
	if acceleration > 0:
	break
	print("The Force is", mass * acceleration)
	
	ii) numbers = [1,2,3,4,5,6,7,8]
	odd_numbers = []
		16	Test 2

	for item in numbers:
	  if item % 2 == 1:
	  odd_numbers.append(item)
	print(odd_numbers)		

Analysis Level			
Unit 1 to 5			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1.	Can we code without design? Justify your answer	2	Test 1
2.	Is feasibility study a necessary one?	2	Test 1
3.	Create Software Requirement Specification (SRS), Use Case diagram; apply any one of the perspective process model.	80	Group Assignment
4.	Create a video on how to use the Junit Tool for testing. Demonstrate a testing case with the tool and post the video in your YouTube channel. Receive comments from your peers and improve your video accordingly.	40	Individual Assignment

Group Assignment			
Unit 1 to 5			
S.NO	**Questions**	**Marks**	**Assessment Tool**
1	Create Software Requirement Specification (SRS), Use Case diagram; apply any one of the perspective process model. Online Shopping	80	Group Assignment
2.	Create Software Requirement Specification (SRS), Use Case diagram; apply any one of the perspective process model. Library Management System	80	Group Assignment
3.	Create Software Requirement Specification (SRS), Use Case diagram; apply any one of the perspective process model. Medical Prescription System	80	Group Assignment