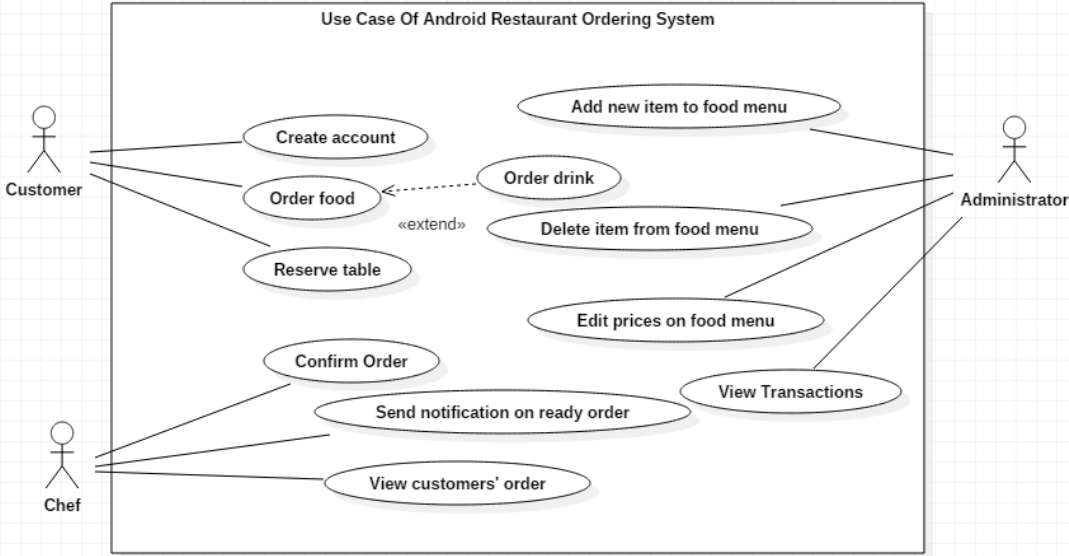


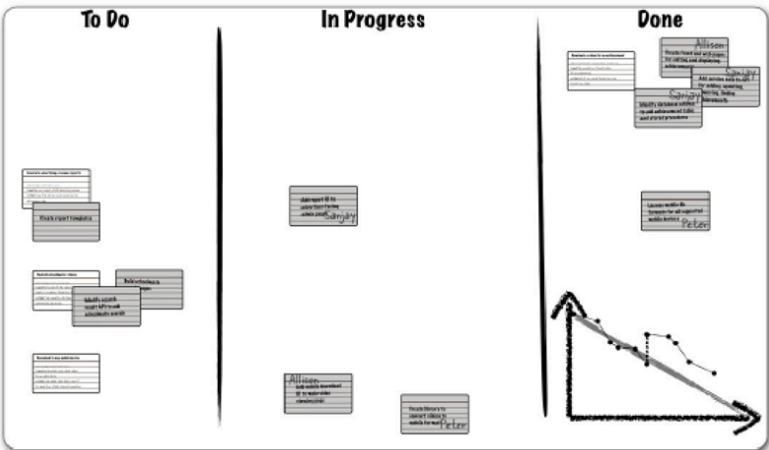
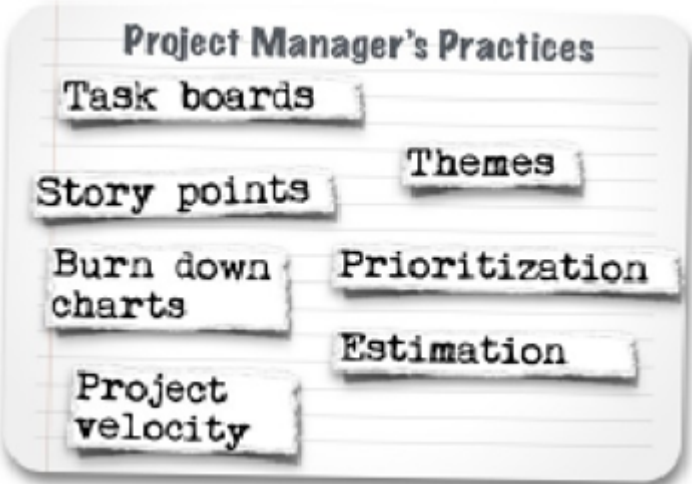
CAT 1 - ANSWER KEY

Course Code : 22CSC51

Course Name : Agile Methodologies

Test Date : 02.09.20224

1.	Identify the collections of a process that are performed when some work products are to be created. Activities Actions Tasks	2
2.	Outline the umbrella activities that occur throughout the software process. Project tracking and control, risk management, quality assurance, configuration management, technical reviews, reusability, documentation and Measurement.	2
3.	Infer the various requirements engineering task. inception, elicitation, elaboration, negotiation, specification, validation, and management	2
4.	List two examples of software projects that would be amenable to the prototyping model on your perspective. Be specific. Custom E-Commerce Website for a Niche Market Healthcare Mobile App for Patient Self-Monitoring Virtual Reality (VR) Training Simulator for Workplace Safety	2
5.	Draw a use case model for activities involved in ordering food in a restaurant from the point when the customer enters into the restaurant to the point when he leaves the restaurant  <pre> graph LR subgraph "Use Case Of Android Restaurant Ordering System" direction TB subgraph Customer C1(Create account) C2(Order food) C3(Reserve table) end subgraph Administrator A1(Add new item to food menu) A2(Delete item from food menu) A3(Edit prices on food menu) A4(View Transactions) end subgraph Chef CH1(Confirm Order) CH2(Send notification on ready order) CH3(View customers' order) end C2 -.-> «extend» OD(Order drink) OD --- A1 OD --- A2 OD --- A3 OD --- A4 end </pre>	2

6.	<p>Prepare the Class-responsibility-collaborator (CRC) card for Bank account class</p> <p>2</p> <table border="1"> <tr> <td colspan="2">Class: Account</td> </tr> <tr> <td>Responsibilities:</td><td>Collaborations:</td> </tr> <tr> <td> Store account holder information (name, address, contact details). Maintain account balance. Perform transactions (deposit, withdrawal). Generate account statements. </td><td> Collaborates with Transaction class for performing deposit and withdrawal operations. Collaborates with Statement class to generate account statements. </td> </tr> </table>	Class: Account		Responsibilities:	Collaborations:	Store account holder information (name, address, contact details). Maintain account balance. Perform transactions (deposit, withdrawal). Generate account statements.	Collaborates with Transaction class for performing deposit and withdrawal operations. Collaborates with Statement class to generate account statements.
Class: Account							
Responsibilities:	Collaborations:						
Store account holder information (name, address, contact details). Maintain account balance. Perform transactions (deposit, withdrawal). Generate account statements.	Collaborates with Transaction class for performing deposit and withdrawal operations. Collaborates with Statement class to generate account statements.						
7.	<p>Represent agile task board with its elements diagrammatically.</p> <p>2</p> 						
8.	<p>Interpret the project manager's practices in agile project development.</p> <p>2</p> 						
9.	<p>Predict some common values of agile manifesto that can lead to an effective team.</p> <p>2</p>						

The Agile Manifesto

Individuals and interactions	over	Processes and Tools
Working Product	over	Comprehensive Documentation
Customer Collaboration	over	Contract Negotiation
Responding to change	over	Following a plan

10. Outline the most popular methodologies that are followed in agile development.

2

Scrum

Kanban

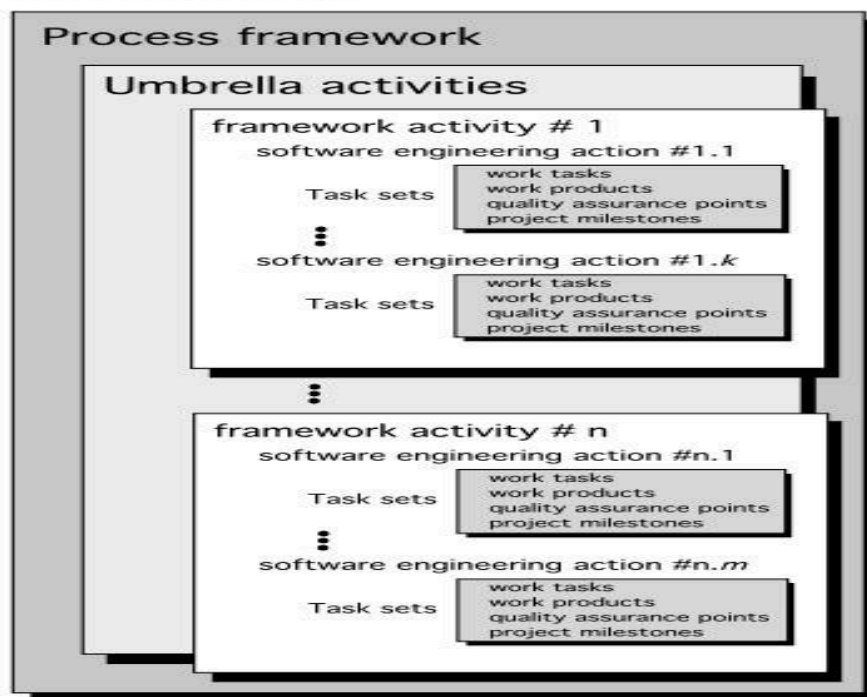
Lean

Extreme Programming

11. Draw the block diagram of a generic process framework and explain its activities.

10

Software process



----- 3 marks

Generic process framework for software engineering defines

Process was defined as a collection of work activities, actions and tasks that are performed when some product is to be created

5 framework activities

----- 3 marks

Communication – Gather Requirements from Customers

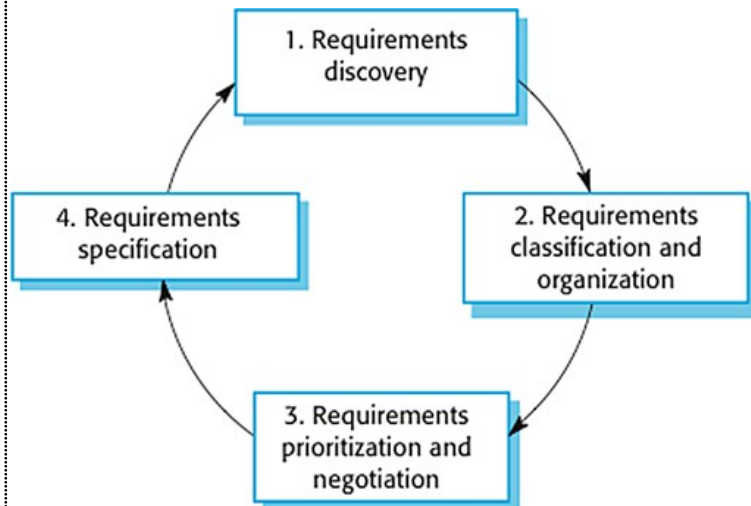
planning – Plan to be followed will be created. It describes the technical task to be conducted, risk ,required resources, work schedule etc.,

		<p>Modeling – A model will be created to better understand the requirements</p> <p>Construction – Code generated and tested</p> <p>Deployment - Complete or partial complete version of software is given to the customer to evaluate and they give feed back based on the evaluation</p> <p>a set of umbrella activities --- 2 marks</p> <p>– Project tracking and control, risk management, quality assurance, configuration management, technical reviews, reusability, documentation and Measurement.</p> <p>A task set defines the actual work to be done to accomplish the objectives of a software engineering action. ---- 2 marks</p> <ul style="list-style-type: none">– A list of the task to be accomplished– A list of the work products to be produced– A list of the quality assurance filters to be applied																																			
12.	a	<p>Compare traditional waterfall model with spiral model</p> <table><tr><th>Aspect</th><th>Waterfall Model</th><th>Spiral Model</th></tr><tr><td>Development Approach</td><td>Linear and sequential</td><td>Iterative with risk analysis</td></tr><tr><td>Phases</td><td>Phases are executed one after the other</td><td>Phases are repeated in spirals (iterations)</td></tr><tr><td>Flexibility</td><td>Rigid, difficult to change requirements</td><td>Flexible, allows for changes during each cycle</td></tr><tr><td>Risk Handling</td><td>Minimal risk assessment</td><td>Extensive risk assessment and mitigation</td></tr><tr><td>Customer Involvement</td><td>Involvement primarily at the beginning</td><td>Ongoing involvement throughout the cycles</td></tr><tr><td>Cost Estimation</td><td>Easier to estimate once requirements are fixed</td><td>Difficult to estimate due to iterative nature</td></tr><tr><td>Best Suited For</td><td>Small, simple projects with well-defined requirements</td><td>Large, complex, high-risk projects</td></tr><tr><td>Iteration</td><td>No iteration; each phase is completed once</td><td>Multiple iterations with refining of the product</td></tr><tr><td>Testing</td><td>Testing occurs after development is complete</td><td>Testing occurs in each iteration</td></tr><tr><td>Risk of Failure</td><td>Higher if requirements are not well understood initially</td><td>Lower due to continuous risk assessment</td></tr></table>	Aspect	Waterfall Model	Spiral Model	Development Approach	Linear and sequential	Iterative with risk analysis	Phases	Phases are executed one after the other	Phases are repeated in spirals (iterations)	Flexibility	Rigid, difficult to change requirements	Flexible, allows for changes during each cycle	Risk Handling	Minimal risk assessment	Extensive risk assessment and mitigation	Customer Involvement	Involvement primarily at the beginning	Ongoing involvement throughout the cycles	Cost Estimation	Easier to estimate once requirements are fixed	Difficult to estimate due to iterative nature	Best Suited For	Small, simple projects with well-defined requirements	Large, complex, high-risk projects	Iteration	No iteration; each phase is completed once	Multiple iterations with refining of the product	Testing	Testing occurs after development is complete	Testing occurs in each iteration	Risk of Failure	Higher if requirements are not well understood initially	Lower due to continuous risk assessment		5
Aspect	Waterfall Model	Spiral Model																																			
Development Approach	Linear and sequential	Iterative with risk analysis																																			
Phases	Phases are executed one after the other	Phases are repeated in spirals (iterations)																																			
Flexibility	Rigid, difficult to change requirements	Flexible, allows for changes during each cycle																																			
Risk Handling	Minimal risk assessment	Extensive risk assessment and mitigation																																			
Customer Involvement	Involvement primarily at the beginning	Ongoing involvement throughout the cycles																																			
Cost Estimation	Easier to estimate once requirements are fixed	Difficult to estimate due to iterative nature																																			
Best Suited For	Small, simple projects with well-defined requirements	Large, complex, high-risk projects																																			
Iteration	No iteration; each phase is completed once	Multiple iterations with refining of the product																																			
Testing	Testing occurs after development is complete	Testing occurs in each iteration																																			
Risk of Failure	Higher if requirements are not well understood initially	Lower due to continuous risk assessment																																			
	b	<p>With a neat sketch, explain in detail about requirement elicitation and analysis</p> <ul style="list-style-type: none">• Analysis of requirements starts with requirement elicitation.• The requirements are analyzed to identify inconsistencies, defects, omission, etc. <p style="text-align: right;">--- 1 mark</p>		5																																	

Requirement Elicitation Techniques

— 3 mark

- Interviews
- Surveys
- Questionnaires
- Task analysis
- Domain Analysis
- Brainstorming
- Prototyping
- Observation

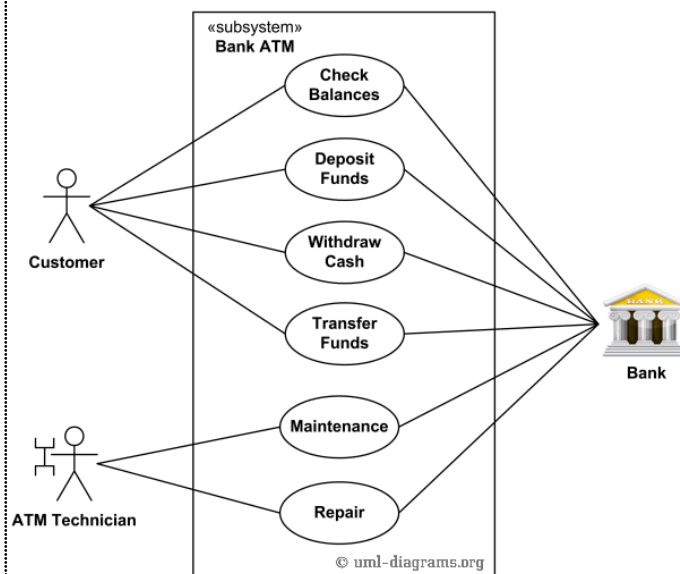


----- 2 marks

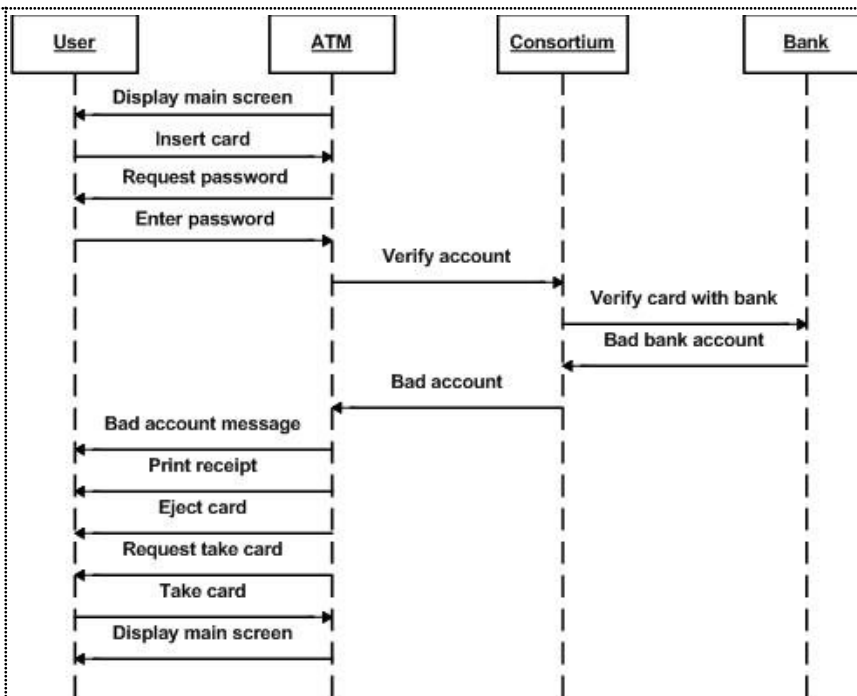
13.

Build the use case model and class diagram for Bank ATM system.

10



-----5 marks



----- 5 mark5

Note:- Consider any other feature included in the use case and sequence diagram

14.

Summarize the 12 principles of agile software

10

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
5. Businesspeople and developers must work together daily throughout the project.
6. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.³