

Internal Assessment Test 1 – June 2024





Sub:	SOFTWARE ENGINEERING & PROJECT MANAGEMENT	SubCode:	21CS61	Branch:	AI&DS					
Date:	03/06/2024	Duration:	90 min's	MaxMarks:	50	Sem/Sec:	VI 'A'	OBE		
Answer any FIVE FULL Questions								MARKS	CO	RBT
1	<p>Indicate why requirement validation is needed. Discuss different check to be carried out during requirement validation process.</p> <p>Need of requirement validation: 2 Marks</p> <p>Different check: 8 Marks</p> <p>Requirements Validation Techniques are used to ensure that the software requirements are complete, consistent, and correct.</p> <p>Requirements validation is the process of checking that requirements defined for development, define the system that the customer wants. To check issues related to requirements, we perform requirements validation. We typically use requirements validation to check errors at the initial phase of development as the error may increase excessive rework when detected later in the development process. In the requirements validation process, we perform a different type of test to check the requirements mentioned in the Software Requirements Specification (SRS), these checks include:</p> <ol style="list-style-type: none">1. Completeness checks2. Consistency checks3. Validity checks4. Realism checks5. Ambiguity checks6. Variability <p>The output of requirements validation is the list of problems and agreed-on actions of detected problems. The lists of problems indicate the problem detected during the process of requirement validation. The list of agreed actions states the corrective action that should be taken to fix the detected problem</p>						10	CO1	L2	
2a)	<p>Define software engineering? Briefly discuss the attributes of good software.</p> <p>Definition: 2 Marks</p> <p>Attributes: 4 Marks</p> <p>Software Engineering is the process of designing, developing, testing, and maintaining software. It is a systematic and disciplined approach to software development that aims to create high-quality, reliable, and maintainable software.</p> <p>Good software quality attributes</p> <ul style="list-style-type: none">• Usability.• Functionality.• Correctness.• Maintainability.• Reliability.• Flexibility.• Scalability.• Testability.						6	CO1	L1	
2b)	<p>Briefly explain the software engineering ethics?</p> <p>Explanation: 4 Marks</p>						4	CO1	L1	

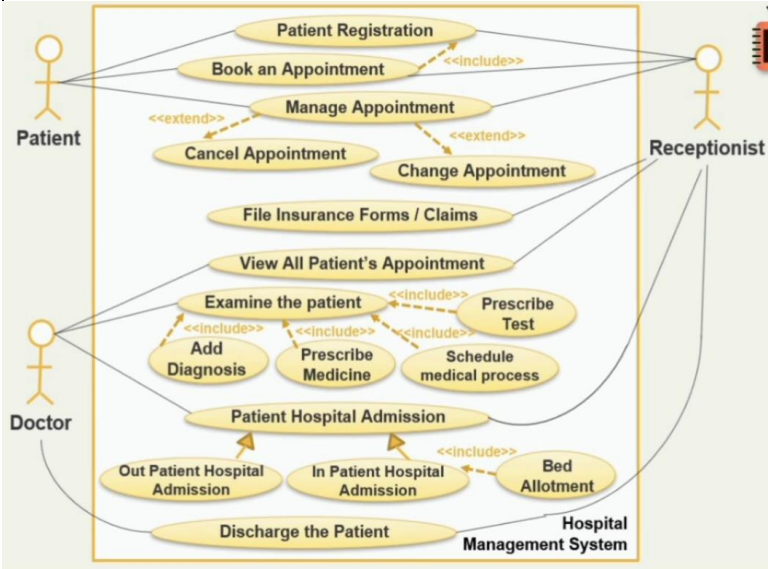
A software engineering code of ethics is a moral compass, guiding professionals to make responsible decisions and ensuring that the products and services they develop align with societal values and expectations. Software engineering is about writing code and creating solutions that positively impact society.

3 What are the components used for use case diagram? Draw use case diagram for hospital management and use all the components in the diagram?

10 CO2 L2

Components: 3 Marks
Use case Diagram: 7 Marks

	NAME	SYMBOL
1.	Use Cases	
2.	Actors	
3.	Association	
4.	System Boundary Boxes	



4 What are the methods needs to build the requirement model.

10 CO1 L2

Methods: 3 Marks
Explanation: 7 Marks

- Scenario-based elements
- Class-based element
- Behavioural elements
- Flow-oriented elements
- Analysis patterns

Scenario-based elements

A scenario-based approach is used to describe the system from the perspective of the user. For example, basic use cases and their related use-case diagrams, evolve into more complicated template-based use cases.

Class-based element

Each usage scenario entails a collection of objects that are modified as an actor

	<p>interacts with the system. These objects are classified as classes— a collection of things with similar characteristics and behaviour.</p> <p>Behavioural elements The behaviour of a computer-based system can have a significant impact on the design and implementation techniques used.</p> <p>Flow-oriented elements As data moves through a computer-based system, it is transformed. The system accepts input in a variety of formats, transforms it using functions, and produces output in a variety of forms.</p> <p>Analysis Patterns Anyone who has done requirements engineering on a number of software projects will note that some issues repeat across all projects within a certain application area. These patterns of analysis provide solutions (e.g., a class, a function, or a behaviour) inside the application domain that can be reused when modelling several applications.</p>			
5	<p>Discuss about concurrent and specialized process models and mention advantages and disadvantages.</p> <p>Concurrent: 5 Marks Specialized process models: 5 Marks</p> <ul style="list-style-type: none"> • It is also called as concurrent process model or concurrent engineering • It defines a series of events that will trigger transition from state to state <div data-bbox="177 981 959 1489" data-label="Diagram"> <p>Analysis activity</p> <pre> graph TD None[None] --> UnderDevelopment[Under development] UnderDevelopment --> AwaitingChanges[Awaiting changes] AwaitingChanges --> UnderRevision[Under revision] UnderRevision --> UnderReview[Under review] UnderReview --> Baseline[Baseline] Baseline --> Done[Done] Done --> UnderDevelopment </pre> <p>Represents a state of a software engineered activity</p> </div> <p>Advantages</p> <ul style="list-style-type: none"> • Applicable to all types of software development process • Easy to understand and use • Gives immediate feedback from testing • Provides an accurate picture of current state of a project. <p>Disadvantages</p> <ul style="list-style-type: none"> • Needs better communication between team members • This may not be achieved all the time <p>Specialized process models</p> <ul style="list-style-type: none"> • Component based development: The process to apply when reuse is a development object. (existing S/W modules) • Formal method: Emphasizes the mathematical specification of requirements • Aspect oriented S/W development: It provides a process and methodological approach for defining, specifying, designing & constructing aspect • Unified process: UML • It helps software developers visualize, construct, and document new software 	10	CO2	L2

systems and blueprints.

6

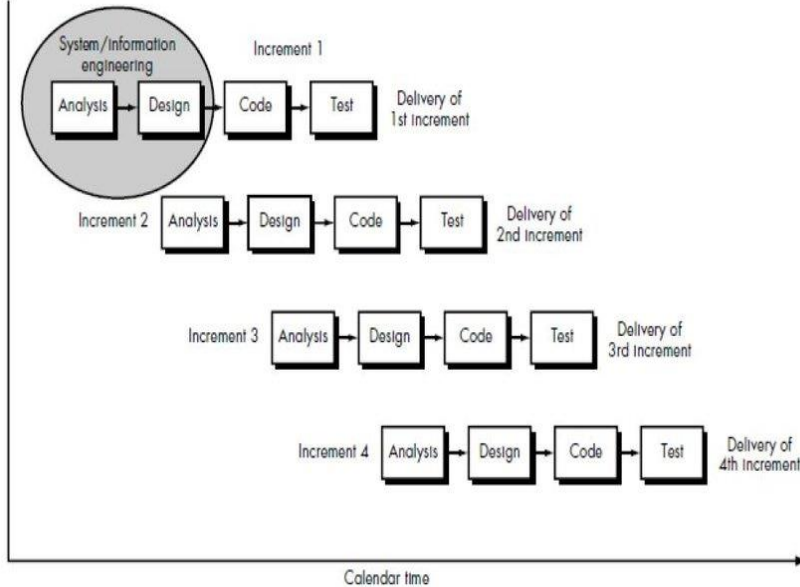
Explain incremental development process model with a neat diagram. Also mention the benefits of this model when compared to waterfall model.

Incremental development process model: 4 Marks

Diagram: 4 Marks

Comparison: 2 Marks

- Requirements are divided into multiple modules
- Each module -> Analysis, design, code & testing
- Used for software with less features
- Less man power is required



In an incremental model large team is not required. In the waterfall model overlapping of phases is not possible. In incremental model overlapping of phases is possible. There is only one cycle in the waterfall model.

10

CO2

L2