Software Engineering

Q.1 (a) [3 Marks]

1. Enlist any two basic components of a system. ***(1 mark)***

- Input
- Output

2. List out the key phases of the Classical SDLC model. ***(1 mark)***

- Requirement Analysis
- System Design
- Implementation
- Testing
- Deployment
- Maintenance

3. Define Software. ***(1 mark)***

- Software is a collection of instructions, data, or programs used to operate computers and perform specific tasks.

Q.1 (b) [7 Marks] ***(1 mark each)***

1. What does RAD stand for?

- **b) Rapid Application Development**

2. True or False: Agile methods seem to work best when team members have a relatively high skill level.

- **True**

3. Which of the following threatens the quality and timeliness of produced software?

- **a) Business risks**

4. In Agile methodology, the _____ is responsible for maximizing the product value and managing the product backlog.

- **Product Owner**

5. The _____ model helps in representing the system's dynamic behavior.

- **State Transition**

6. _____ is not a direct measure of SE process.

- **a) Effort**

7. In a Data Flow Diagram (DFD), a _____ represents the transformation of data from input to output.

- **Process**

Q.2 (a) [4 Marks]

1. Explain why the Waterfall model is suitable for developing Airplane Flight Control Software. (2 marks)

- Waterfall model is suitable due to its sequential nature. Airplane software requires strict testing, well-defined stages, and minimal changes during development, all supported by the Waterfall model.

2. Discuss any two Software Application Domains in brief. (2 marks)

- **Healthcare Systems**: Software for managing patient records, appointments, and diagnostics.
- **Banking Systems**: Applications for online banking, transaction processing, and fraud detection.

Q.2 (b) [6 Marks]

1. Describe the Classical SDLC and its phases. (3 marks)

- **Phases:**
    1. **Requirement Gathering**
    2. **System Design**
    3. **Implementation**
    4. **Testing**
    5. **Deployment**
    6. **Maintenance**
- This model follows a linear, step-by-step approach ideal for clearly defined projects.

2. Define Process and discuss process framework activities. (3 marks)

- **Process**: A set of related tasks to achieve a specific goal.
- **Framework Activities**:
    - Communication
    - Planning
    - Modeling
    - Construction
    - Deployment

Q.3 Attempt Any TWO [10 Marks]

1. Requirement Engineering and Example. (5 marks)

- **Definition**: Process of gathering, analyzing, documenting, and validating software requirements.
- **Phases**:
    - Elicitation
    - Specification
    - Validation
    - Management
- **Example**: For a railway reservation system, requirements include seat availability, payment integration, and user login.

2. Compare Prototype and RAD Model. (5 marks)

Feature	Prototype Model	RAD Model
Focus	Building prototype for feedback	Rapid development with reuse
Speed	Medium	High
Cost	Medium	High (due to tools and manpower)
End User Involvement	High	Very High
Suitable for	Systems requiring user interfaces	Projects needing quick delivery

3. Define SRS and create SRS for Online Food Ordering System. (5 marks)

- **SRS (Software Requirement Specification)**: A detailed description of software system to be developed, laying out functional and non-functional requirements.

SRS for Online Food Ordering System:

- **Functional Requirements**:
    - User Registration/Login
    - Browse Menu
    - Add to Cart
    - Place Order
    - Payment Gateway Integration
- **Non-functional Requirements**:
    - Secure transactions
    - User-friendly interface
    - System uptime of 99.9%

Q.4 (a) [5 Marks]

Create an E-R Diagram and Decision Tree for Employee Leave Management

Entities:

- Employee (Emp_ID, Name)
- Leave (Leave_ID, Type, Duration)
- Manager (Mgr_ID, Name)

Relationships:

- Employee *applies for* Leave
- Manager *approves* Leave

Decision Tree (sample):

- If employee has leave balance → Yes → Manager approval?
    - Yes → Approve Leave
    - No → Reject Leave
- Else → Reject Leave

Q.4 (b) [5 Marks]

Option 1: Adaptive Software Development Process (with figure)

- **Phases**:
    1. Speculation – initial planning
    2. Collaboration – iterative development with stakeholders
    3. Learning – review outcomes and adapt
- It focuses on rapid adaptation, learning, and collaboration.

OR

Option 2: Scrum Process Model (with figure)

- **Roles**:
    - Product Owner
    - Scrum Master
    - Development Team
- **Artifacts**:
    - Product Backlog
    - Sprint Backlog
    - Increment

- **Events**:
    - Sprint Planning
    - Daily Scrum
    - Sprint Review & Retrospective