Unit – I

1. Which of the following is NOT a software process model?

   - A) Waterfall model

   - B) Incremental model

   - C) Spiral model

   - D) Modular model

   Answer: D

2. The Spiral model was proposed by:

   - A) Royce

   - B) Boehm

   - C) Pressman

   - D) Sommerville

   Answer: B

3. The Waterfall model is best suited for:

   - A) Large, complex projects with well-defined requirements

   - B) Projects with high risk and changing requirements

   - C) Projects with tight deadlines

   - D) None of the above

   Answer: A

4. Which model is characterized by short iterative cycles and quick releases?

   - A) Waterfall model

   - B) Incremental model

   - C) Agile model

   - D) V-model

 Answer: C

5. In the context of software engineering, the term "process model" refers to:

   - A) A set of tools for coding

   - B) A way to create a software product

   - C) A standardized format for documentation

   - D) A strategy for testing software

Answer: B

6. Which of the following is NOT a characteristic of good software?

   - A) Maintainability

   - B) Usability

   - C) Complexity

   - D) Reliability

Answer: C

7. Portability in software refers to:

   - A) The ability to use the software on multiple hardware platforms

   - B) The ease of moving the software from one place to another

   - C) The software's ability to recover from failures

   - D) The ability to integrate with other software

   Answer: A

8. What does "robustness" in software imply?

   - A) Ease of use

   - B) Efficient performance

   - C) Ability to handle errors gracefully

   - D) Compatibility with other systems

   Answer: C

9. Which of the following is a quality attribute of software?

   - A) Functionality

   - B) Modularity

   - C) Documentation

   - D) Redundancy

Answer: A

10. The degree to which software can be understood, corrected, adapted, and enhanced is referred to as:

   - A) Usability

   - B) Reliability

   - C) Maintainability

   - D) Efficiency

   Answer: C

11. Which of the following is NOT considered a software component?

   - A) Modules

   - B) Libraries

   - C) Servers

   - D) Functions

   Answer: C

12. What is the main advantage of using software components?

   - A) Increased cost

   - B) Improved performance

   - C) Reusability

   - D) Complexity

   Answer: C

13. The process of assembling software from pre-existing components is known as:

   - A) Integration

   - B) Composition

   - C) Compilation

   - D) Linkage

   Answer: B

14. Which component model is used extensively in Windows operating systems?

   - A) JavaBeans

   - B) CORBA

   - C) COM

   - D) .NET

   Answer: C

15. Components that interact via interfaces to achieve a common goal in software systems are known as:

   - A) Classes

   - B) Objects

   - C) Modules

   - D) Components

   Answer: D

16. Application software is designed to:

   - A) Operate hardware

   - B) Provide a platform for other software

   - C) Help the user perform specific tasks

   - D) Manage system resources

   Answer: C

17. Which of the following is an example of system software?

   - A) Word processor

   - B) Operating system

   - C) Web browser

   - D) Spreadsheet

   Answer: B

18. An embedded application is typically found in:

   - A) Desktop computers

   - B) Mainframes

   - C) Mobile devices

   - D) Microwave ovens

   Answer: D

19. The primary purpose of application software is to:

   - A) Support the computer's basic functions

   - B) Perform tasks for the user

   - C) Manage network resources

   - D) Protect against malware

 Answer: B

20. Software designed for end-users to perform a specific task is called:

   - A) Middleware

   - B) System software

   - C) Utility software

   - D) Application software

 Answer: D

21. Which layer of software engineering focuses on tools and methods for software development?

   - A) Application layer

   - B) Platform layer

   - C) Process layer

   - D) Methods layer

   Answer: D

22. The process layer in software engineering encompasses:

   - A) Programming tools

   - B) The activities, actions, and tasks required to build high-quality software

   - C) User requirements

   - D) System hardware

   Answer: B

23. At which layer do technologies like compilers and debuggers reside?

   - A) Tools layer

   - B) Process layer

   - C) Methods layer

   - D) Quality layer

   Answer: A

24. The application layer in software engineering is concerned with:

   - A) User interfaces

   - B) Business logic

   - C) Data storage

   - D) Network protocols

   Answer: A

25. Which of the following is NOT part of the layered technology in software engineering?

   - A) Process layer

   - B) Methods layer

   - C) Quality layer

   - D) Debugging layer

   Answer: D

26. Software processes involve:

   - A) A sequence of steps required to develop software

   - B) Tools used for coding

   - C) Methods for documenting software

   - D) Standards for quality assurance

   Answer: A

27. Methods in software engineering are:

   - A) Techniques for coding

   - B) Structured approaches to solving software engineering problems

   - C) Tools for project management

   - D) Processes for quality assurance

   Answer: B

28. Tools in software engineering typically refer to:

   - A) Hardware components

   - B) Software programs that support the software development process

   - C) Methods for documenting software

   - D) Standards for quality assurance

   Answer: B

29. An example of a software engineering tool is:

   - A) An algorithm

   - B) A design method

   - C) A compiler

   - D) A process model

   Answer: C

30. The primary goal of using processes, methods, and tools in software engineering is to:

   - A) Increase complexity

   - B) Reduce cost

   - C) Improve productivity and quality

   - D) Standardize documentation

   Answer: C

31. The generic view of software engineering includes:

   - A) Requirements analysis, design, coding, testing, and maintenance

   - B) Just coding and testing

   - C) Only maintenance

   - D) Documentation only

   Answer: A

32. Which phase is NOT part of the generic software engineering framework?

   - A) Coding

   - B) Design

   - C) Marketing

   - D) Testing

   Answer: C

33. The primary objective of the maintenance phase in software engineering is to:

   - A) Develop new software

   - B) Correct faults, improve performance, and adapt to a changed environment

   - C) Perform testing

   - D) Design the architecture

   Answer: B

34. Requirements analysis in software engineering involves:

- A) Determining user needs and documenting them

- B) Writing code

- C) Testing software

- D) Maintaining software

Answer: A

35. The design phase in software engineering primarily focuses on:

- A) Coding

- B) Identifying user needs

- C) Developing a blueprint for the software solution

- D) Testing software

Answer: C

36. The Waterfall model is also known as:

- A) Linear sequential model

- B) Iterative model

- C) Agile model

- D) Prototype model

Answer: A

37. In the Waterfall model, each phase must be completed:

   - A) Simultaneously

   - B) Before the next phase begins

   - C) Independently of the others

   - D) In any order

   Answer: B

38. A major disadvantage of the Waterfall model is:

   - A) It is not suitable for large projects

   - B) It does not handle changing requirements well

   - C) It is too flexible

   - D) It lacks structure

Unit – II

1. What is the primary goal of software project management?

   - A) To write code

   - B) To manage risks

   - C) To ensure that the project is completed on time, within budget, and meets the required quality

   - D) To document the software requirements

   Answer: C

2. Which of the following is NOT a phase in software project management?

   - A) Initiation

   - B) Planning

   - C) Coding

   - D) Closing

   Answer: C

3. The management spectrum includes:

   - A) People, Product, Process, Project

   - B) Plan, Do, Check, Act

   - C) Requirements, Design, Implementation, Testing

   - D) Scope, Time, Cost, Quality

   Answer: A

4. Which principle states that project objectives must be defined explicitly?

   - A) W5HH Principle

   - B) SMART Goals

   - C) MoSCoW Prioritization

   - D) FURPS Model

   Answer: A

5. The W5HH principle includes questions that cover:

   - A) Who, What, When, Where, Why, and How

   - B) Which, When, Where, How, How much

   - C) Why, What, When, Who, How

   - D) Why, What, Which, When, How

   Answer: C

6. In the management spectrum, which element is considered the most important?

   - A) Product

   - B) People

   - C) Process

   - D) Project

   Answer: B

7. Which of the following is NOT a factor in managing people in a software project?

   - A) Motivation

   - B) Team structure

   - C) Programming language

   - D) Communication

   Answer: C

8. What is a key focus when managing the 'product' in software project management?

   - A) Defining the product's scope and objectives

   - B) Ensuring team members are motivated

   - C) Monitoring project schedules

   - D) Risk management

   Answer: A

9. The 'process' component of the management spectrum focuses on:

   - A) Selecting the right team

   - B) The activities and tasks required to produce the product

   - C) Project timelines

   - D) Product features

   Answer: B

10. Effective project management ensures that:

   - A) The project is completed at the lowest cost possible

   - B) The team follows the exact steps regardless of circumstances

   - C) The project meets its goals and objectives

   - D) The project avoids any form of documentation

   Answer: C

11. The 'People' aspect in software project management focuses on:

   - A) Hardware resources

   - B) Team dynamics and individual capabilities

   - C) Software tools

   - D) Project timelines

   Answer: B

12. The 'Product' aspect in software project management involves:

   - A) Hardware configurations

   - B) Defining the deliverables and the requirements

   - C) Team roles and responsibilities

   - D) Testing strategies

   Answer: B

13. The 'Process' aspect in software project management is primarily concerned with:

   - A) Writing code

   - B) Following a defined set of activities and tasks

   - C) Team member selection

   - D) Client communication

   Answer: B

14. The 'Project' aspect in software project management refers to:

   - A) The final product delivered to the client

   - B) The collection of all activities, resources, and timelines needed to achieve the project objectives

   - C) The budget allocated for the project

   - D) The user requirements

   Answer: B

15. The success of the 'People' aspect largely depends on:

   - A) The tools they use

   - B) Their motivation, skills, and ability to work as a team

   - C) The defined processes

   - D) The project budget

   Answer: B

16. Who developed the W5HH principle?

   - A) Barry Boehm

   - B) Watts Humphrey

   - C) Frederick Brooks

   - D) Roger Pressman

   Answer: D

17. The "Why" in the W5HH principle addresses:

   - A) Project objectives and goals

   - B) Team roles

   - C) Project schedules

   - D) Resource allocation

   Answer: A

18. The "What" in the W5HH principle focuses on:

   - A) Project timelines

   - B) Defining the work products and deliverables

   - C) Risk management

   - D) Team selection

   Answer: B

19. In the W5HH principle, "When" pertains to:

   - A) The timeline and milestones of the project

   - B) The project's budget

   - C) The tools to be used

   - D) The project's scope

   Answer: A

20. The "How" in the W5HH principle is about:

   - A) How the project will be managed

   - B) How the final product will be marketed

   - C) How the team will be motivated

   - D) How the risks will be mitigated

   Answer: A

21. Effective team management in a software project leads to:

   - A) Higher costs

   - B) Reduced productivity

   - C) Increased collaboration and project success

   - D) Frequent conflicts

   Answer: C

22. A key characteristic of a successful software team is:

  - A) Homogeneity

  - B) Diversity of skills and perspectives

  - C) Strict hierarchy

  - D) Independent work

  Answer: B

23. Which of the following is a challenge in team management?

  - A) Clear communication

  - B) Lack of diversity

  - C) High motivation levels

  - D) Defined roles

  Answer: B

24. Team building activities aim to:

  - A) Increase competition among team members

  - B) Improve teamwork and collaboration

  - C) Create individual silos

  - D) Minimize interaction

  Answer: B

25. Effective communication in a team ensures:

   - A) Individual work is prioritized over team goals

   - B) Misunderstandings and errors are minimized

   - C) Work is done independently

   - D) Deadlines are frequently missed

   Answer: B

26. The first step in planning a software project is:

   - A) Writing code

   - B) Project scheduling

   - C) Scope definition and feasibility analysis

   - D) Quality planning

   Answer: C

27. Scope definition in a project primarily involves:

   - A) Listing project deliverables

   - B) Estimating project costs

   - C) Scheduling tasks

   - D) Identifying team members

   Answer: A

28. A feasibility study in project planning assesses:

   - A) Technical, economic, legal, operational, and schedule feasibility

   - B) Only the economic feasibility

   - C) Only the technical feasibility

   - D) Only the legal feasibility

   Answer: A

29. Effort estimation is crucial for:

   - A) Identifying stakeholders

   - B) Determining the project's timelines and resources needed

   - C) Defining project scope

   - D) Developing the software code

   Answer: B

30. Which technique is commonly used for effort estimation?

   - A) SWOT analysis

   - B) Gantt charts

   - C) Function Point Analysis

   - D) PERT charts

   Answer: C

31. The COCOMO model is used for:

   - A) Risk management

   - B) Effort estimation

   - C) Quality assurance

   - D) Team building

   Answer: B

32. What does the acronym COCOMO stand for?

   - A) Comprehensive Cost Modeling

   - B) Constructive Cost Model

   - C) Computerized Cost Management

   - D) Coordinated Cost Methodology

   Answer: B

33. Function Point Analysis measures:

   - A) Lines of code

   - B) The complexity of the software

   - C) The functionality delivered to the user

   - D) The number of functions in the software

   Answer: C

34. In effort estimation, historical data is useful because:

   - A) It provides exact estimates for future projects

   - B) It helps in making informed estimates based on past projects

   - C) It eliminates the need for expert judgment

   - D) It guarantees project success

   Answer: B

35. Which of the following is NOT a factor considered in the COCOMO model?

   - A) Product attributes

   - B) Hardware attributes

   - C) Personnel attributes

   - D) Marketing

Answer: D

Unit – III

1. Requirements engineering is primarily concerned with:

   - A) Coding

   - B) Defining and managing the requirements of a software system

   - C) Testing

   - D) Maintenance

   Answer: B

2. What is the first step in requirements engineering?

   - A) Requirements validation

   - B) Requirements specification

   - C) Problem recognition

   - D) Requirements analysis

   Answer: C

3. Which of the following is NOT a task in requirements engineering?

   - A) Requirements elicitation

   - B) Requirements analysis

   - C) Requirements specification

   - D) Software deployment

   Answer: D

4. Requirements engineering involves stakeholders such as:

   - A) Developers

   - B) End-users

   - C) Project managers

   - D) All of the above

   Answer: D

5. The output of the requirements engineering process is:

   - A) The source code

   - B) The requirements specification document

   - C) The test cases

   - D) The user manual

   Answer: B

6. Problem recognition in requirements engineering refers to:

   - A) Identifying the technical challenges in a project

   - B) Understanding the actual needs and issues faced by the stakeholders

   - C) Defining the budget for the project

   - D) Creating a prototype

   Answer: B

7. During problem recognition, which of the following techniques is often used?

   - A) Code reviews

   - B) Brainstorming sessions

   - C) Unit testing

   - D) Deployment

   Answer: B

8. The goal of problem recognition is to:

   - A) Develop the software architecture

   - B) Identify and understand the problem domain

   - C) Write the code

   - D) Test the software

   Answer: B

9. Which of the following is an important activity during problem recognition?

   - A) Coding

   - B) Stakeholder analysis

   - C) Deployment

   - D) Debugging

   Answer: B

10. Problem recognition helps in:

    - A) Reducing the complexity of the code

    - B) Aligning the project with stakeholder needs and expectations

    - C) Increasing the project budget

    - D) Designing the database schema

    Answer: B

11. Which of the following is NOT a requirement engineering task?

   - A) Requirements elicitation

   - B) Requirements analysis

   - C) Requirements specification

   - D) Software coding

   Answer: D

12. Requirements elicitation involves:

   - A) Identifying and gathering requirements from stakeholders

   - B) Writing the source code

   - C) Performing unit tests

   - D) Deploying the software

   Answer: A

13. The process of refining and detailing the gathered requirements is called:

   - A) Requirements elicitation

   - B) Requirements analysis

   - C) Requirements specification

   - D) Requirements validation

   Answer: B

14. Requirements specification results in:

   - A) The final software product

   - B) A detailed documentation of the requirements

   - C) A set of test cases

   - D) The project deployment plan

   Answer: B

15. Requirements validation ensures that:

- A) The requirements are correctly implemented in the code

- B) The requirements accurately reflect the needs of the stakeholders

- C) The software is free of bugs

- D) The project is completed on time

Answer: B

16. Which of the following is a common process in requirements engineering?

- A) Requirements elicitation

- B) Software testing

- C) Debugging

- D) Software deployment

Answer: A

17. The requirements engineering process is typically:

- A) Linear

- B) Iterative

- C) Ad-hoc

- D) Unstructured

Answer: B

18. Requirements elicitation techniques include:

- A) Prototyping

- B) Brainstorming

- C) Interviews

- D) All of the above

Answer: D

19. Which process involves checking the requirements for feasibility, consistency, and completeness?

   - A) Requirements elicitation

   - B) Requirements analysis

   - C) Requirements specification

   - D) Requirements validation

   Answer: D

20. The main objective of the requirements engineering process is to:

   - A) Develop a detailed design of the software

   - B) Ensure that the software meets the needs and expectations of the stakeholders

   - C) Write the source code

   - D) Perform system testing

   Answer: B

21. The requirements specification document is also known as:

   - A) Software design document

   - B) User manual

   - C) Software requirements specification (SRS)

   - D) Test plan

   Answer: C

22. Which of the following is NOT typically included in a requirements specification document?

   - A) Functional requirements

   - B) Non-functional requirements

   - C) Source code

   - D) System constraints

   Answer: C

23. Functional requirements describe:

   - A) The behavior of the system

   - B) The performance of the system

   - C) The design of the system

   - D) The coding standards

   Answer: A

24. Non-functional requirements specify:

   - A) What the system should do

   - B) How the system should perform

   - C) The user interfaces

   - D) The database schema

   Answer: B

25. Which of the following is an example of a non-functional requirement?

   - A) The system shall allow users to log in

   - B) The system shall respond to user inputs within 2 seconds

   - C) The system shall generate monthly reports

   - D) The system shall support multiple user roles

   Answer: B

26. A use case represents:

   - A) A detailed description of a user's interaction with the system

   - B) The internal structure of the system

   - C) The database schema

   - D) The user interface design

   Answer: A

27. Use cases are primarily used for:

   - A) Requirements elicitation

   - B) Requirements validation

   - C) Requirements specification

   - D) All of the above

   Answer: D

28. The main components of a use case are:

   - A) Actors, system, scenarios

   - B) Tables, fields, queries

   - C) Classes, objects, methods

   - D) Servers, clients, networks

   Answer: A

29. Functional specifications are concerned with:

   - A) The technical implementation of the system

   - B) The behavior and operations of the system

   - C) The project's budget

   - D) The hardware requirements

   Answer: B

30. A use case diagram typically includes:

   - A) Use cases and actors

   - B) Classes and objects

   - C) Data flow diagrams

   - D) Entity-relationship diagrams

   Answer: A

31. Requirements validation is conducted to:

   - A) Develop the software architecture

   - B) Ensure that the requirements are correct and complete

   - C) Perform unit testing

   - D) Deploy the software

   Answer: B

32. Which technique is NOT used for requirements validation?

   - A) Prototyping

   - B) Review meetings

   - C) Simulation

   - D) Code inspection

   Answer: D

33. During requirements validation, the requirements are checked for:

   - A) Consistency and completeness

   - B) Performance and scalability

   - C) Implementation details

   - D) Database schema

   Answer: A

34. One of the main goals of requirements validation is to:

   - A) Ensure that the software meets the requirements of the stakeholders

   - B) Develop the detailed design of the system

   - C) Write the source code

   - D) Perform system testing

   Answer: A

35. Which of the following is an activity performed during requirements validation?

   - A) Unit testing

   - B) Requirements review

   - C) Code inspection

   - D) Database design

   Answer: B

36. Requirements analysis aims to:

   - A) Develop a solution to the identified problem

   - B) Write the project documentation

   - C) Understand and document the requirements in detail

   - D) Perform user acceptance testing

   Answer: C

37. Which technique is commonly used in requirements analysis?

   - A) Use case modeling

   - B) Debugging

   - C) System integration

   - D) Code review

   Answer: A

38. During requirements analysis, requirements are:

   - A) Developed and validated

   - B) Defined and prioritized

   - C) Implemented and tested

 - D) Documented and archived

   Answer: B

39. What is a key challenge in requirements analysis?

   - A) Writing code

   - B) Ensuring the requirements are clear, complete, and consistent

   - C) Testing the software

   - D) Deploying the software

   Answer: B

40. Which of the following is NOT a requirement analysis technique?

   - A) Use case modeling

   - B) Entity-relationship diagrams

   - C) Data flow diagrams

   - D) System deployment

   Answer: D

41. Which method focuses on user stories and scenarios to gather requirements?

   - A) Agile requirements gathering

   - B) Waterfall model

   - C) Spiral model

   - D) V-Model

   Answer: A

42. In Agile requirements engineering, the requirements are:

   - A) Fixed and detailed upfront

   - B) Flexible and evolving

   - C) Defined by the project manager alone

   - D) Static and unchangeable

   Answer: B

43. Which of the following is a tool used for managing requirements?

   - A) JIRA

   - B) Git

   - C) Docker

   - D) Jenkins

   Answer: A

44. Traceability in requirements engineering refers to:

   - A) Tracking the source code changes

   - B) Tracking the origin and evolution of each requirement

   - C) Tracking the testing phases

   - D) Tracking the deployment process

   Answer: B

45. Requirements prioritization is important because:

   - A) It helps in deciding which requirements to implement first

   - B) It eliminates the need for testing

   - C) It reduces the project timeline

   - D) It increases the complexity of the project

   Answer: A

46. Which of the following is a requirements prioritization technique?

   - A) MoSCoW method

   - B) Bubble sort

   - C) Binary search

   - D) Merge sort

   Answer: A

47. The MoSCoW method stands for:

   - A) Must have, Should have, Could have, Won't have

   - B) Must be, Should be, Can be, Won't be

   - C) Might be, Should be, Could be, Won't be

   - D) Must be, Shall be, Could be, Won't be

   Answer: A

48. Requirements negotiation is needed when:

   - A) There are conflicting requirements from different stakeholders

   - B) The requirements are clear and well-defined

   - C) The project scope is fixed

   - D) The budget is unlimited

   Answer: A

49. Which of the following is a benefit of effective requirements management?

   - A) Reduced project scope

   - B) Increased customer satisfaction

   - C) Decreased team communication

   - D) Reduced project timeline

   Answer: B

50. Requirement management tools help in:

   - A) Coding the software

   - B) Tracking, managing, and maintaining requirements

   - C) Testing the software

   - D) Deploying the software

   Answer: B

51. Which role is primarily responsible for gathering requirements?

   - A) Software Developer

   - B) Business Analyst

   - C) Database Administrator

   - D) Network Engineer

   Answer: B

52. Requirements elicitation can be challenging due to:

   - A) Communication barriers

   - B) Lack of stakeholder involvement

   - C) Vague and conflicting requirements

   - D) All of the above

   Answer: D

53. Prototyping helps in:

   - A) Understanding and refining requirements through a visual model

   - B) Writing the final code

   - C) Testing the software

   - D) Deploying the software

   Answer: A

54. Requirements traceability is useful for:

   - A) Ensuring each requirement is implemented and tested

   - B) Managing project timelines

   - C) Reducing project costs

   - D) Eliminating the need for documentation

   Answer: A

55. A requirement that specifies a constraint on the system is a:

   - A) Functional requirement

   - B) Non-functional requirement

   - C) Business requirement

   - D) Technical requirement

   Answer: B

56. Which of the following is NOT a non-functional requirement?

   - A) Performance

   - B) Usability

   - C) Reliability

   - D) User authentication

   Answer: D

57. Requirements modeling involves:

   - A) Creating abstract representations of the requirements

   - B) Writing the source code

   - C) Conducting integration tests

   - D) Deploying the system

   Answer: A

58. Which diagram is often used in requirements modeling to represent user interactions?

   - A) Class diagram

   - B) Use case diagram

   - C) Sequence diagram

   - D) Component diagram

   Answer: B

59. Requirements validation helps in:

  - A) Reducing the risk of missing critical requirements

  - B) Writing the detailed design document

  - C) Coding the software

  - D) Performing system tests

  Answer: A

60. Requirements analysis helps to:

  - A) Identify any inconsistencies or gaps in the requirements

  - B) Finalize the source code

  - C) Test the software

  - D) Deploy the software

   Answer: A

61. Which of the following is an elicitation technique involving group sessions?

  - A) Interviews

  - B) Brainstorming

  - C) Document analysis

  - D) Surveys

  Answer: B

62. Document analysis in requirements engineering involves:

  - A) Reviewing existing documentation to identify requirements

  - B) Writing new documentation

  - C) Coding the requirements

  - D) Testing the documentation

   Answer: A

63. Requirements workshops are useful because they:

   - A) Encourage collaboration and consensus among stakeholders

   - B) Limit stakeholder involvement

   - C) Focus on coding

   - D) Reduce project budget

   Answer: A

64. Which of the following helps in prioritizing requirements?

   - A) Use case modeling

   - B) Prototyping

   - C) MoSCoW analysis

   - D) Data flow diagrams

   Answer: C

65. Which of the following is a common issue with poorly managed requirements?

   - A) Increased stakeholder satisfaction

   - B) Scope creep

   - C) Decreased project timeline

   - D) Reduced development costs

   Answer: B

66. Requirements management tools typically provide:

   - A) Version control for requirements

   - B) Real-time collaboration

   - C) Traceability and tracking

   - D) All of the above

   Answer: D

67. Stakeholder interviews help in:

 - A) Gathering detailed and specific requirements

 - B) Writing the final code

 - C) Testing the software

 - D) Designing the database

 Answer: A

68. Use cases are often accompanied by:

 - A) Use case narratives

 - B) Source code

 - C) Test cases

 - D) Deployment scripts

 Answer: A

69. Requirements reviews typically involve:

 - A) A systematic examination of the requirements document by a team

 - B) Writing the source code

 - C) Performing unit tests

 - D) Deploying the system

  Answer: A

70. The goal of requirements negotiation is to:

 - A) Resolve conflicts and agree on a common set of requirements

 - B) Write the detailed design

 - C) Test the software

 - D) Deploy the software

 Answer: A

71. Volatile requirements are those that:

   - A) Change frequently

   - B) Are always stable

   - C) Are never implemented

   - D) Are low priority

   Answer: A

72. Which of the following is a dynamic elicitation technique?

   - A) Observation

   - B) Prototyping

   - C) Document analysis

   - D) Questionnaires

   Answer: B

73. Requirement patterns help by:

   - A) Providing reusable solutions to common requirements problems

   - B) Writing the source code

   - C) Testing the software

   - D) Reducing the need for stakeholder involvement

   Answer: A

74. Conflict management in requirements engineering involves:

   - A) Resolving disagreements between stakeholders about requirements

   - B) Coding the requirements

   - C) Testing the software

   - D) Deploying the system

   Answer: A

75. Requirements metrics help in:

   - A) Measuring the quality and progress of requirements engineering activities

   - B) Writing the final code

   - C) Testing the software

   - D) Reducing the project budget

   Answer: A

76. Requirements that cannot be changed once agreed upon are called:

   - A) Frozen requirements

   - B) Volatile requirements

   - C) Flexible requirements

   - D) Baseline requirements

   Answer: D

77. Scenario-based analysis focuses on:

   - A) Detailed narratives describing how users will interact with the system

   - B) The database design

   - C) The software architecture

   - D) The deployment plan

   Answer: A

78. Requirements dependency tracking is important for:

   - A) Understanding how changes in one requirement may affect others

   - B) Writing the source code

   - C) Testing the software

   - D) Reducing the project scope

   Answer: A

79. Which of the following helps in visualizing the relationships between requirements?

   - A) Requirements traceability matrix

   - B) Source code

   - C) Unit tests

   - D) Deployment scripts

   Answer: A

80. Change control in requirements engineering refers to:

   - A) Managing changes to the requirements after they have been approved

   - B) Writing the final code

   - C) Testing the software

   - D) Deploying the system

   Answer: A

81. Which of the following helps in ensuring that all requirements are addressed during development?

   - A) Traceability matrix

   - B) Coding standards

   - C) Design patterns

   - D) Testing scripts

   Answer: A

82. Requirements validation can be performed through:

   - A) Prototyping and reviews

   - B) Coding and testing

   - C) Debugging and deploying

   - D) Documenting and archiving

   Answer: A

83. Use case scenarios are useful because they:

   - A) Provide a step-by-step description of user interactions

   - B) Define the database schema

   - C) Describe the network architecture

   - D) Outline the coding standards

   Answer: A

84. Requirements gathering sessions with multiple stakeholders are called:

   - A) Focus groups

   - B) Stand-up meetings

   - C) Code reviews

   - D) System tests

   Answer: A

85. Requirements engineering involves:

   - A) Eliciting, analyzing, specifying, validating, and managing requirements

   - B) Writing the source code

   - C) Performing system tests

   - D) Deploying the software

   Answer: A

86. Which technique involves users interacting with a working model of the system to gather requirements?

   - A) Prototyping

   - B) Interviews

   - C) Surveys

   - D) Brainstorming

   Answer: A

87. Requirement conflicts are best resolved through:

- A) Stakeholder negotiation and prioritization

- B) Ignoring low-priority requirements

- C) Immediate coding and testing

- D) Delaying the project timeline

Answer: A

88. Requirements feasibility analysis evaluates:

- A) The practicality of implementing the requirements within given constraints

- B) The correctness of the source code

- C) The performance of the system

- D) The deployment strategy

Answer: A

89. A requirement that describes a feature of the system is a:

- A) Functional requirement

- B) Non-functional requirement

- C) Project requirement

- D) Budget requirement

Answer: A

90. In Agile development, requirements are often captured as:

- A) User stories

- B) Detailed specification documents

- C) Database schemas

- D) System architecture diagrams

Answer: A

91. Requirements validation can involve:

   - A) Stakeholder reviews and acceptance testing

   - B) Writing detailed design documents

   - C) Performing unit tests

   - D) System deployment

   Answer: A

92. The requirements traceability matrix is used to:

   - A) Track the implementation of each requirement throughout the project lifecycle

   - B) Define the project budget

   - C) Design the user interface

   - D) Perform system tests

   Answer: A

93. Agile methodologies emphasize:

   - A) Continuous requirements gathering and iteration

   - B) Detailed upfront requirements specification

   - C) Strict adherence to initial requirements

   - D) Minimal stakeholder involvement

   Answer: A

94. Requirements prioritization helps in:

   - A) Focusing on the most critical requirements first

   - B) Reducing the project timeline

   - C) Eliminating the need for validation

   - D) Avoiding stakeholder involvement

   Answer: A

95. Functional requirements are typically captured in:

   - A) Use cases and user stories

   - B) System architecture diagrams

   - C) Database schemas

   - D) Network configuration documents

   Answer: A

96. Non-functional requirements can include:

   - A) Performance, scalability, and security

   - B) Specific features and functionalities

   - C) User interfaces and interactions

   - D) Database table structures

   Answer: A

97. The primary goal of requirements engineering is to:

   - A) Ensure the software meets the needs and expectations of its users

   - B) Write the source code

   - C) Perform system testing

   - D) Deploy the software

   Answer: A

98. Elicitation techniques like surveys and questionnaires are useful for:

   - A) Gathering a large amount of information from many stakeholders

   - B) Writing the final code

   - C) Testing the software

   - D) Designing the database

   Answer: A

99. Requirements workshops typically result in:

   - A) A clearer and more detailed understanding of stakeholder needs

   - B) Finalized source code

   - C) Completed system tests

   - D) The deployment of the system

   Answer: A

100. Effective requirements management helps in:

   - A) Reducing the risk of project failure

   - B) Eliminating the need for testing

   - C) Increasing the project budget

   - D) Decreasing stakeholder involvement

   Answer: A