Multiple Choice Questions and Answers

1. Which of the following is a non-functional requirement?
    - A) The system shall display the user's profile after login
    - B) **The system shall process 10,000 transactions per second**
    - C) The system shall allow users to reset their password
    - D) The system shall support user authentication

2. Which of the following is NOT an advantage of Agile methodology?
    - A) Increased flexibility
    - B) Faster delivery of working software
    - C) **Extensive upfront documentation**
    - D) Better responsiveness to change

3. In software testing, unit testing is concerned with:
    - A) Testing the entire system
    - B) **Testing individual modules or components**
    - C) Testing the system’s integration with external systems
    - D) Testing the system’s performance under load

4. Which software development model is also called the “linear sequential model”?
    - A) Spiral Model
    - B) **Waterfall Model**
    - C) Agile Model
    - D) V-Model

5. Which of the following is an example of white-box testing?
    - A) Regression testing
    - B) Equivalence partitioning
    - C) **Code coverage analysis**
    - D) Usability testing

Answers to Open Questions

6. **What is the difference between functional and non-functional requirements?**
   Functional requirements specify what the system should do (e.g., user authentication), while non-functional requirements define how the system performs a function (e.g., performance, usability).

7. **How does the waterfall model differ from the agile model?**
   The waterfall model is a linear and sequential approach where each phase must be completed before moving to the next, whereas the agile model is iterative and incremental, promoting flexibility and continuous improvement.

8. **What is the purpose of software testing, and what are its main levels?**
   Software testing aims to identify and fix defects to ensure quality and functionality. The main levels are unit testing, integration testing, system testing, and acceptance testing.

9. **What are the characteristics of good software?**
   Good software is functional, reliable, efficient, user-friendly, maintainable, and scalable.

10. **How do version control systems help in software development?**
    Version control systems manage changes to source code, enabling collaboration, history tracking, and rollback to previous versions when necessary.

11. **What is the role of a software architect in a development team?**
    A software architect designs the system's structure, ensuring it meets technical and business requirements, and oversees the integration of software solutions.

12. **What is the difference between verification and validation?**
    Verification checks if the product is built correctly according to the design specifications, while validation ensures the product meets user needs and requirements.

13. **What are software design patterns, and why are they important?**
    Design patterns are reusable solutions to common software design problems, providing a proven template to improve code efficiency and maintainability.

14. **What are the main challenges of software maintenance?**
    Challenges include managing legacy code, ensuring compatibility, handling evolving requirements, and maintaining documentation.

15. **How does continuous integration (CI) improve software quality?**
    CI automates code integration and testing, allowing frequent and reliable updates, reducing integration issues, and improving code quality.

16. **What is technical debt, and how can it affect a project?**
    Technical debt is the implied cost of additional rework caused by choosing an easy solution now instead of a better approach. It can lead to increased maintenance costs and reduced development speed.

17. **What are the advantages of using microservices over monolithic architecture?**
    Microservices offer improved scalability, flexibility, and resilience, allowing independent deployment and development of services.

18. **What is the purpose of requirement elicitation?**
    Requirement elicitation aims to gather, analyze, and document the needs and expectations of stakeholders for a software project.

19. **How does user-centered design influence software development?**
    User-centered design focuses on the end-users' needs and preferences, ensuring the software is intuitive, accessible, and user-friendly.

20. **What are the trade-offs between time, cost, and quality in software projects?**
    Balancing these factors involves prioritizing project constraints, where improving one aspect may impact the others, requiring careful management and negotiation.

21. **What is the difference between black-box and white-box testing?**
    Black-box testing evaluates the software's functionality without knowledge of the internal code, while white-box testing involves testing the internal structures and workings.

22. **What are the key principles of the SOLID design principles?**
    The SOLID principles include Single