Free Questions for **CSQE** by **dumpshq**

Shared by **Hayden** on **12-06-2024**

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Which of the following lifecycle models would be most appropriate to use if the requirements are well known?

Options:

A- spiral

B- iterative

C- waterfall

D- incremental

Answer:

C

Explanation:

The waterfall lifecycle model is most appropriate to use when the requirements are well known and unlikely to change. This model follows a linear and sequential approach where each phase must be completed before the next one begins. Because it assumes stable and well-defined requirements, it allows for thorough planning and design upfront, which can lead to a more structured and predictable

development process. Waterfall is less flexible in accommodating changes compared to iterative or agile models, making it suitable for projects with clear, stable requirements. This approach is detailed in traditional software engineering textbooks and standards such as the IEEE Standard for Developing Software Life Cycle Processes (IEEE 1074).

Question 2

Which of the following is the best derived requirement based upon the requirement below? "The user interface needs to be accessible to the visually impaired.

Options:

A- The user interface has a joystick

B- The user interface has a keyboard.

C- The user interface has HD resolution

D- The user interface has voice activation.

Answer:

D

Explanation:

For a requirement stating that the user interface needs to be accessible to the visually impaired, the best-derived requirement is that the user interface has voice activation. Voice activation directly addresses the needs of visually impaired users by allowing them to interact with the software using voice commands instead of relying on visual cues. This makes the interface more accessible and aligns with accessibility standards such as the Web Content Accessibility Guidelines (WCAG) and Section 508 of the Rehabilitation Act, which mandate that technology be accessible to people with disabilities.

Question 3

Question Type: MultipleChoice

Which of the following metrics would be most appropriate for evaluating how effective the peer review process was for removing defects?

Options:

A- Cycle time

B- Defect density

C- First-pass yield

D- Requirements volatility

Answer:

B

Explanation:

Defect density is a metric that measures the number of defects relative to the size of the software component (e.g., per thousand lines of code). It is particularly useful for evaluating the effectiveness of the peer review process in removing defects. By calculating defect density before and after the peer review process, teams can assess how many defects were identified and addressed during reviews. This metric helps in understanding the quality of the code and the effectiveness of the review process in identifying and mitigating defects. Higher effectiveness in peer reviews typically results in lower defect density. Industry practices and standards, such as those from IEEE or ISO, often recommend using defect density as a key quality metric.

Question 4

Question Type: MultipleChoice

Test-driven development in extreme programming ensures the quality of code by

Options:

A- defining test pass criteria before code is written

B- defining what code modules will be tested

C- having the testers determine the order in which code should be written

D- having the code tested before the product is released

Answer:

A

Explanation:

Test-driven development (TDD) is a key practice in extreme programming (XP) that ensures the quality of code by defining the test pass criteria before any code is written. In TDD, developers write automated test cases before writing the actual code. These tests specify what the code should do and serve as a guide for development. The process follows a cycle of writing a test, running it to see it fail (since the code hasn't been written yet), writing the minimum code necessary to pass the test, and then refactoring the code while ensuring the test still passes. This approach helps ensure that code meets the required specifications from the outset and encourages writing only the necessary code to pass tests, leading to higher quality and more maintainable code. Reference for TDD practices can be found in books like 'Test-Driven Development: By Example' by Kent Beck.

Question 5

Which of the following activities is part of performing verification and validation on safety-critical features at even' step of the life cycle?

Options:

A- Identifying and mitigating security risks to prevent safety and hazard risks

B- Documenting software safer.' risk plans that include rigorous development processes

C- Performing software hazard analysis and determining acceptable levels of safety risks

D- Providing mechanisms for discovering, correcting, and preventing the recurrence of safety' risks

Answer:

C

Explanation:

Performing verification and validation on safety-critical features involves several key activities aimed at ensuring that software functions correctly and safely throughout its lifecycle. One critical activity is performing software hazard analysis and determining acceptable levels of safety risks. This process involves identifying potential hazards that the software could pose and evaluating the risks associated with these hazards. By assessing these risks, teams can establish acceptable safety levels and implement measures to mitigate or eliminate these risks. This activity is fundamental in ensuring that the software meets safety requirements and standards, such as those outlined in safety-critical industry guidelines like ISO 26262 for automotive or DO-178C for aerospace.

Question 6

Question Type: **MultipleChoice**

Which of the following techniques would be most appropriate for identifying the source of a problem?

Options:

A- Affinity diagram

B- Root cause analysis

C- Prioritization matrix

D- Interrelationship digraph

Answer:

B

Explanation:

Root cause analysis (RCA) is a systematic process used to identify the underlying causes of a problem. It is the most appropriate technique for identifying the source of a problem due to the following reasons:

Problem Identification: RCA focuses on identifying the fundamental issues that lead to a problem, rather than just addressing the symptoms.

Systematic Approach: It involves a structured methodology to investigate and analyze the problem, ensuring that all potential causes are considered.

Effective Solutions: By understanding the root causes, organizations can implement effective and long-lasting solutions to prevent recurrence.

Question 7

Question Type: MultipleChoice

During concurrent development., a user reported a problem with Version 1 of the software product. A software quality engineer determined the problem originated m Version 2 of the product. Which of the following actions would be most appropriate for the software quality engineer to take first to fix the problem?

Options:

A- Merge the fix into Version 3.

B- Branch the Version 2 codeline

C- Add new functionality to Version 2.

D- Label Version 2 as part of Version 1

Answer:

B

Explanation:

In a concurrent development environment, when a problem is identified in an earlier version (Version 1) that originated in a later version (Version 2), the appropriate action involves managing the code versions effectively. Branching the Version 2 codeline is the most appropriate first step because:

Isolation of Changes: Branching allows for the isolation of the problematic code in Version 2, enabling targeted fixes without affecting ongoing development in other versions.

Concurrent Development: This approach supports concurrent development activities, allowing different teams to work on fixing the issue in Version 2 while continuing development on other versions.