



BCS

Exam Questions CTFL4

ISTQB Certified Tester Foundation Level CTFL 4.0 Exam

About ExamBible

Your Partner of IT Exam

Found in 1998

ExamBible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, ExamBible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

All examinations will be up to date.

* 24/7 Quality Support

We will provide service round the clock.

* 100% Pass Rate

Our guarantee that you will pass the exam.

* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1

Which of the statements correctly describes when a whole team approach may NOT be suitable?

- A. When a high level of test independence may be required.
- B. When acceptance tests need to be created.
- C. When a test automation approach needs to be determined.
- D. When the team dynamics need to be improved.

Answer: A

Explanation:

The whole team approach involves collaboration among all team members, including testers, developers, and business representatives, to achieve quality goals. However, this approach may not be suitable in situations where a high level of test independence is required. Test independence is essential in cases where unbiased testing is critical, such as in regulated environments or where high-risk systems are involved. This is because team members might unintentionally influence each other's work, leading to potential bias in testing outcomes.

NEW QUESTION 2

Which of the following statements about traceability is FALSE?

- A. Traceability between test basis items and the test cases designed to cover them, makes it possible to determine which test basis items have been covered by the executed test cases.
- B. Traceability between test basis items and the test cases designed to cover them, enables experience-based test techniques to be applied
- C. Traceability between test basis items and the test cases designed to cover them, enables identification of which test cases will be affected by changes to the test basis items.
- D. Traceability can be established and maintained through all the test documentation for a given test level, such as from test conditions through test cases to test scripts.

Answer: B

Explanation:

Traceability primarily refers to the ability to link test cases back to their sources in the test basis, such as requirements or design documents. This linkage allows for the determination of coverage, impact analysis, and maintaining consistency across test documentation. However, traceability does not directly enable the application of experience-based test techniques, which are more about using the tester's intuition and experience. The ISTQB CTFL Syllabus v4.0 does not state that traceability enables experience-based techniques, making option B the false statement.

NEW QUESTION 3

The acceptance criteria associated with a user story:

- A. are often written in a rule-oriented format using the template referred to as "Given/When/Then"
- B. are often documented following in rule-oriented format using the following template: "As a [role], I want [feature], so that I can [benefit]"
- C. can be written in different formats and represent an aspect of a user story referred to as confirmation' of the so called "3 C's"
- D. must be written in one of the two following formats: scenario-oriented or rule-oriented

Answer: C

Explanation:

The acceptance criteria associated with a user story are the conditions that must be met for the user story to be considered done and to deliver the expected value to the user. They are often written in different formats, such as rule-oriented, scenario-oriented, or table-oriented, depending on the nature and complexity of the user story. They represent an aspect of a user story referred to as confirmation, which is one of the so called "3 C's" of user stories. The other two aspects are card and conversation. Card refers to the concise and informal description of the user story, usually following the template: "As a [role], I want [feature], so that I can [benefit]". Conversation refers to the ongoing dialogue between the stakeholders and the team members to clarify and refine the user story and its acceptance criteria. Therefore, option C is the correct answer.

References: ISTQB® Certified Tester Foundation Level Syllabus v4.01, Section 3.2.2, page 35-36; ISTQB® Glossary v4.02, page 37.

NEW QUESTION 4

Which of the following characterizations applies to a test tool used for the analysis of a developer's code prior to its execution?

- A. Tool support for test design and implementation.
- B. Tool support for static testing.
- C. Tool support for test execution and logging.
- D. Tool support for performance measurement and dynamic analysis.

Answer: B

Explanation:

A test tool used for the analysis of a developer's code prior to its execution falls under the category of static testing tools. Static testing involves examining the code and documentation without executing the code. These tools are used to perform static analysis, which helps in identifying potential defects and code quality issues early in the development process. The ISTQB CTFL syllabus specifies that static analysis tools are essential for finding defects that do not manifest themselves during the execution of the program.

References: ISTQB CTFL Syllabus, Section 3.1, "Static Testing."

NEW QUESTION 5

Which of the following is not an example of a typical generic skill required for testing?

- A. Be able to apply test-driven development
- B. Be able to use test management tools and defect tracking tools
- C. Be able to communicate defects and failures to developers as objectively as possible

D. Possess the necessary social skills that support effective teamwork

Answer: A

Explanation:

Test-driven development is not an example of a typical generic skill required for testing, but rather an example of a specific technical skill or a development practice that may or may not be relevant for testing, depending on the context and the objectives of the testing activities. Test-driven development is an approach to software development and testing, in which the developers write automated unit tests before writing the source code, and then refactor the code until the tests pass. Test-driven development can help to improve the quality, the design, and the maintainability of the code, as well as to provide fast feedback and guidance for the developers. However, test-driven development is not a skill that is generally expected or needed for testers, especially for testers who are not involved in unit testing or who do not have access to the source code. The other options are examples of typical generic skills required for testing, which are skills that are applicable and beneficial for testing in any context or situation, regardless of the specific testing techniques, tools, or methods used. The typical generic skills required for testing include:

- ? Be able to use test management tools and defect tracking tools: These are tools that help testers to plan, organize, monitor, and control the testing activities and resources, as well as to record, track, analyze, and resolve the defects detected during testing. These tools can improve the efficiency, the effectiveness, and the communication of the testing process, as well as to provide traceability, metrics, and reports for the testing outcomes.
- ? Be able to communicate defects and failures to developers as objectively as possible: This is a skill that involves the ability to report and describe the defects and failures found during testing in a clear, concise, accurate, and unbiased manner, using relevant information, evidence, and terminology, without making assumptions, judgments, or accusations. This skill can facilitate the collaboration, the understanding, and the resolution of the defects and failures between the testers and the developers, as well as to prevent conflicts, misunderstandings, or blame games.
- ? Possess the necessary social skills that support effective teamwork: These are skills that involve the ability to interact, cooperate, and coordinate with other people involved in or affected by the testing activities, such as the test manager, the test team, the project manager, the developers, the customers, the users, etc. These skills can include communication, negotiation, leadership, motivation, feedback, conflict resolution, etc. These skills can enhance the quality, the productivity, and the satisfaction of the testing process, as well as to foster a positive and constructive testing culture. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.1.1, Testing and the Software Development Lifecycle
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.1.2, Testing and Quality
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.2.1, Testing Principles
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.2.2, Testing Policies, Strategies, and Test Approaches
 - ? ISTQB® Glossary of Testing Terms v4.0, Test-driven Development, Test Management Tool, Defect Tracking Tool, Defect Report, Failure, Social Skill2

NEW QUESTION 6

Which of the following statements is true?

- A. Experience-based test techniques rely on the experience of testers to identify the root causes of defects found by black-box test techniques
- B. Some of the most common test basis used by white-box test techniques include user stories, use cases and business processes
- C. Experience-based test techniques are often useful to detect hidden defects that have not been targeted by black-box test techniques
- D. The primary goal of experience-based test techniques is to design test cases that can be easily automated using a GUI-based test automation tool

Answer: C

Explanation:

Experience-based test techniques are test design techniques that rely on the experience, knowledge, intuition, and creativity of the testers to identify and execute test cases that are likely to find defects in the software system. Experience-based test techniques are often useful to detect hidden defects that have not been targeted by black-box test techniques, which are test design techniques that use the external behavior and specifications of the software system as the test basis, without considering its internal structure or implementation. Experience-based test techniques can complement black-box test techniques by covering aspects that are not explicitly specified, such as usability, security, reliability, performance, etc. The other statements are false, because:

- ? Experience-based test techniques do not rely on the experience of testers to identify the root causes of defects found by black-box test techniques, but rather to identify the potential sources of defects based on their own insights, heuristics, or exploratory testing. The root causes of defects are usually identified by debugging or root cause analysis, which are activities that involve examining the code or the development process to find and fix the errors that led to the defects.
- ? Some of the most common test basis used by white-box test techniques include the source code, the design documents, the architecture diagrams, and the control flow graphs of the software system. White-box test techniques are test design techniques that use the internal structure and implementation of the software system as the test basis, and aim to achieve a certain level of test coverage based on the code elements, such as statements, branches, paths, etc. User stories, use cases, and business processes are examples of test basis used by black-box test techniques, as they describe the functional and non-functional requirements of the software system from the perspective of the users or the stakeholders.
- ? The primary goal of experience-based test techniques is not to design test cases that can be easily automated using a GUI-based test automation tool, but rather to design test cases that can reveal defects that are not easily detected by other test techniques, such as boundary value analysis, equivalence partitioning, state transition testing, etc. Test automation is the use of software tools to execute test cases and compare actual results with expected results, without human intervention. Test automation can be applied to different types of test techniques, depending on the test objectives, the test levels, the test tools, and the test resources. However, test automation is not always feasible or beneficial, especially for test cases that require human judgment, creativity, or exploration, such as those designed by experience-based test techniques. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.1, Black-box Test Design Techniques
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.2, White-box Test Design Techniques
 - ? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.3, Experience-based Test Design Techniques
 - ? ISTQB® Glossary of Testing Terms v4.0, Experience-based Test Technique, Black-box Test Technique, White-box Test Technique, Test Basis, Test Coverage, Test Automation

NEW QUESTION 7

Which of the following statements about statement coverage is TRUE?

- A. Achieving 90% statement coverage ensures that 90% branch coverage is achieved.
- B. Achieving 100% statement coverage ensures that no variable within the code has been used without being initialised.
- C. Achieving 100% statement coverage ensures that 100% branch coverage is achieved
- D. Achieving 80% statement coverage ensures that 80% of all executable statements within the code have been exercised.

Answer: D

Explanation:

Statement coverage measures the percentage of executable statements that have been exercised by a test suite. Achieving 80% statement coverage means that

80% of the executable code lines have been tested. This metric helps in understanding how much of the code has been covered during testing. However, it does not guarantee branch coverage, variable initialization, or detection of all possible defects. The ISTQB CTFL Syllabus v4.0 explains statement coverage as a measure of the extent to which the code has been tested, without implying other types of coverage or testing goals.

NEW QUESTION 8

A financial institution is to implement a system that calculates the interest rates paid on investment accounts based on the sum invested. You are responsible for testing the system and decide to use equivalence partitioning and boundary value analysis to design test cases. The requirements describe the following expectations:

Investment range| Interest rate
R500 to R1000| 10%

R1001 to R5000| 11%
R5001 to R10000| 12%
R10001 to R50000| 13%

What is the minimum number of test cases required to cover all valid equivalence partitions for calculating the interest?

- A. 5
- B. 4
- C. 8
- D. 16

Answer: B

Explanation:

Using equivalence partitioning, the investment ranges are divided into four partitions:

? R500 to R10,000 (10%)

? R10,001 to R50,000 (11%)

? R50,001 to R100,000 (12%)

? R100,001 to R500,000 (13%)

Thus, the minimum number of test cases required to cover all valid equivalence partitions for calculating the interest is 4.

NEW QUESTION 9

Which of the following statements about white-box test techniques is true?

- A. Achieving full statement coverage and full branch coverage for a software product means that such software product has been fully tested and there are no remaining bugs within the code
- B. Code-related white-box test techniques are not required to measure the actual code coverage achieved by black-box testing, as code coverage can be measured using the coverage criteria associated with black-box test techniques
- C. Branch coverage is the most thorough code-related white-box test technique, and therefore applicable standards prescribe achieving full branch coverage at the highest safety levels for safety-critical systems
- D. Code-related white-box test techniques provide an objective measure of coverage and can be used to complement black-box test techniques to increase confidence in the code

Answer: D

Explanation:

This answer is correct because code-related white-box test techniques are test design techniques that use the structure of the code to derive test cases. They provide an objective measure of coverage, such as statement coverage, branch coverage, or path coverage, which indicate how much of the code has been exercised by the test cases. Code-related white-box test techniques can be used to complement black-box test techniques, which are test design techniques that use the functional or non-functional requirements of the system or component to derive test cases. By combining both types of techniques, testers can increase their confidence in the code and find more defects. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.3.2.2

NEW QUESTION 10

A test status report SHOULD:

- A. Specify the impediments to carrying out the planned test activities in the reporting period and the corresponding solutions put in place to remove them
- B. Be produced as part of test completion activities and report unmitigated product risks to support the decision whether or not to release the product
- C. Always be based on the same template within an organisation, as its structure and contents should not be affected by the audience to which the report is presented.
- D. Specify the lines of communication between testing, other lifecycle activities, and within the organisation that were chosen at the outset of the test project.

Answer: A

Explanation:

A test status report is a document that provides a snapshot of the testing activities and their progress during a particular period. It should include information about any impediments encountered during the test execution and the actions taken to resolve them, which helps stakeholders understand the challenges and how they were addressed.

Option B describes an activity related to test completion rather than ongoing status reporting. Option C is incorrect because the structure and contents of the report may vary based on the audience's needs. Option D, while important, is not the primary purpose of a test status report, which focuses more on the current status and impediments.

NEW QUESTION 10

What is test oracle?

- A. The source of test objectives
- B. The source for the actual results
- C. The source of expected results
- D. The source of input conditions

Answer: C

Explanation:

A test oracle is a mechanism or principle that can be used to determine whether the observed behavior or output of a system under test is correct or not. A test oracle can be based on various sources of expected results, such as specifications, user expectations, previous versions, comparable systems, etc. References: ISTQB Certified Tester Foundation Level(CTFL) v4.0 Syllabus, Section 1.2.1, Page 91; ISTQB Glossary of Testing Terms, Version 4.0, Page 332.

NEW QUESTION 14

The following rules determine the annual bonus to be paid to a salesman of a company based on the total annual amount of the sales made (referred to as TAS). If the TAS is between 50k€ and 80k€, the bonus is 10%. If the TAS exceeds 80k€ by a value not greater than 40k€, the bonus is 15%. Finally, if the TAS exceeds the maximum threshold which entitles to a 15% bonus, the bonus is 22%. Consider applying equivalence partitioning to the TAS (Note: 1k€ = 1000 euros). Which one of the following answers contain only test cases that belong to the same equivalence partition?

- A. TC1 = 81 k€; TC2= 97k€; TC3=111k€; TC4=118k€
- B. TC1 = 40k€; TC2= 46k€; TC3=51k€; TC4=53k€
- C. TC1 = 79k€; TC2= 80k€; TC3=81k€; TC4=82k€
- D. TC1 = 90k€; TC2= 110k€; TC3=125k€; TC4=140k€

Answer: A

Explanation:

This answer is correct because equivalence partitioning is a test design technique that divides the input domain of a system or component into partitions of equivalent data, such that each partition is expected to produce the same output or behavior. Equivalence partitioning aims to reduce the number of test cases by selecting one representative value from each partition. In this case, the input domain of the TAS can be divided into four partitions based on the bonus rules: less than 50k€, between 50k€ and 80k€, between 80k€ and 120k€, and more than 120k€. The test cases in the answer belong to the same partition, which is between 80k€ and 120k€, and they are expected to produce the same output, which is a bonus of 15%. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.3.2.1

NEW QUESTION 17

Consider a review for a high-level architectural document written by a software architect. The architect does most of the review preparation work, including distributing the document to reviewers before the review meeting. However, reviewers are not required to analyze the document in advance, and during the review meeting the software architect explains the document step by step. The only goal of this review is to establish a common understanding of the software architecture that will be used in a software development project. Which of the following review types does this review refer to?

- A. Inspection
- B. Audit
- C. Walkthrough
- D. Informal review

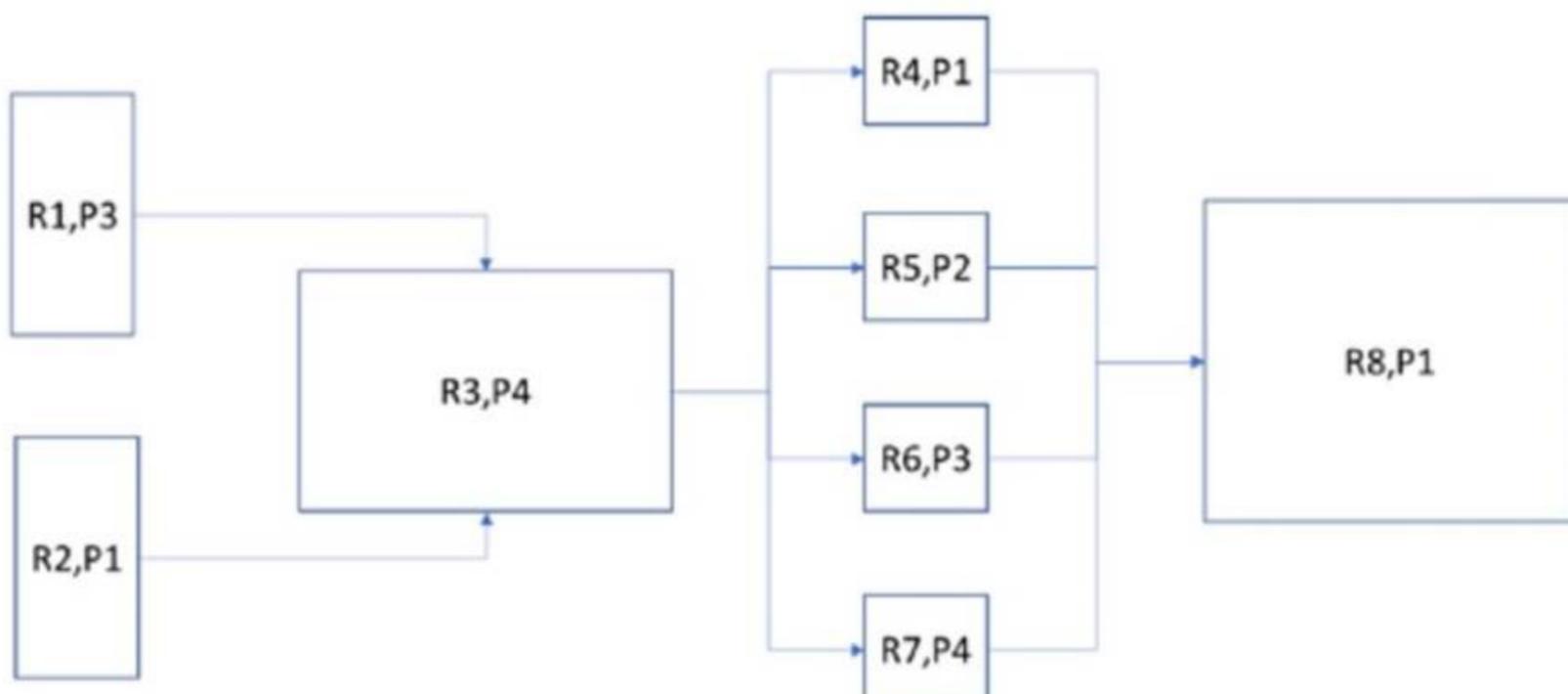
Answer: C

Explanation:

This answer is correct because a walkthrough is a type of review where the author of the work product leads the review process and explains the work product to the reviewers. The reviewers are not required to prepare for the review in advance, and the main objective of the walkthrough is to establish a common understanding of the work product and to identify any major defects or issues. A walkthrough is usually informal and does not follow a defined process or roles. In this case, the review for a high-level architectural document written by a software architect matches the characteristics of a walkthrough. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.4.2.2

NEW QUESTION 22

The following diagram displays the logical dependencies between requirements and the individual requirement priorities. For example, "R2->R3" means that R3 is dependent on R2. Priority is indicated by the number next to the letter "P" i.e. P1 has a higher priority than P2.



Which one of the following options best describes the test execution sequence using both requirement dependency and priority

- A. R2, R1, R3, R4, R5, R6, R7, R8.
- B. R1, R2, R3, R4, R5, R6, R7, R8.
- C. R2, R4, R8, R5, R1, R6, R3, R7.

D. . R2, R1,R3,R7,R6,R5,R4,R8.

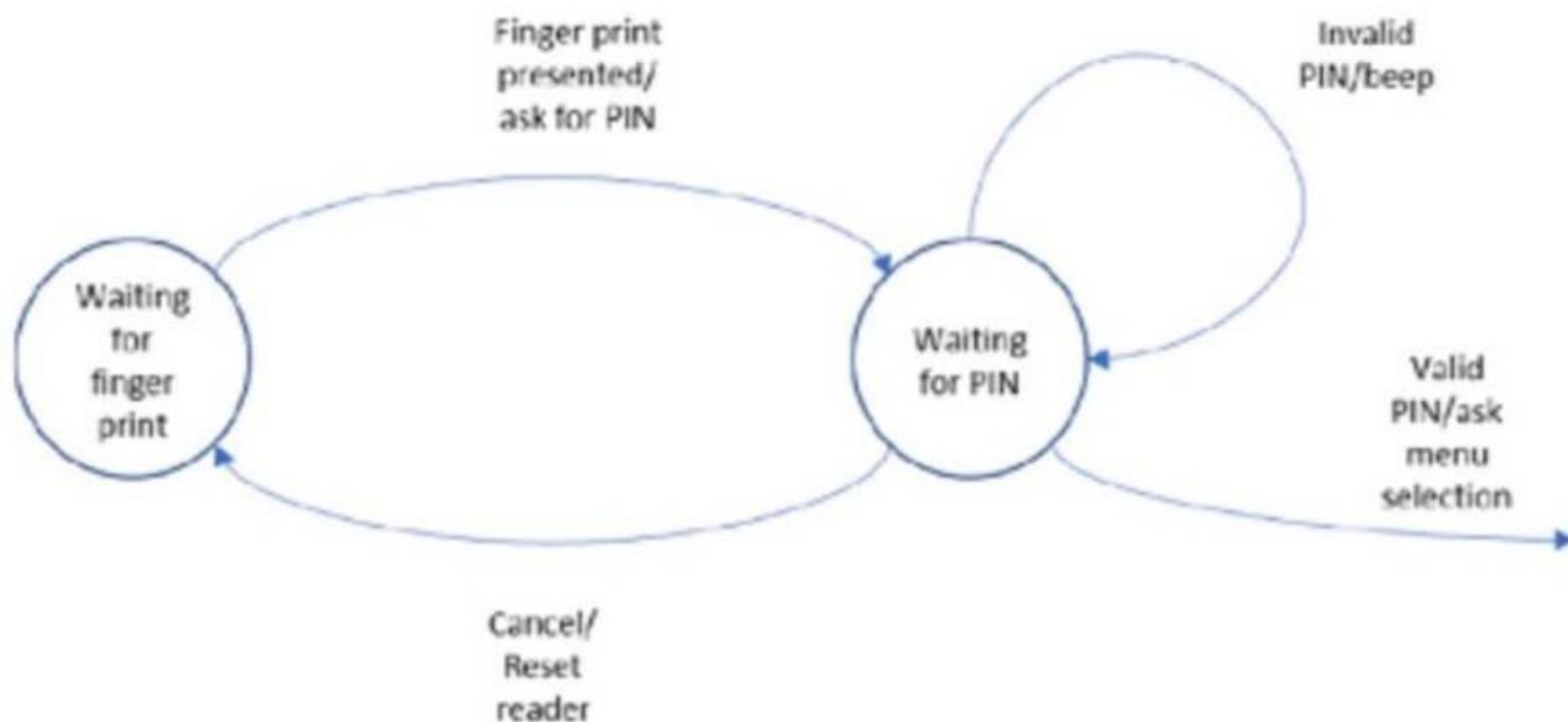
Answer: D

Explanation:

The correct test execution sequence should consider both the dependencies between the requirements and their priorities. According to the diagram, the sequence begins with R2 (P1) as it is a prerequisite for R3 (P4). Then R1 (P3) can be tested. R3 follows as it depends on R2. Next, R7 (P4) should be tested before R6 (P3) and R5 (P2), as indicated by their dependencies. Finally, R4 (P1) and R8 (P1) can be tested. Therefore, the best sequence is R2, R1, R3, R7, R6, R5, R4, R8. Reference:ISTQB CTFL Syllabus V4.0, Section 5.1.5

NEW QUESTION 25

The following state transition diagram describes the functionality involved in a system using fingerprint and password authentication to log onto a system.



How many distinct states of the system are visible in the above diagram?

- A. 1
- B. 2
- C. 3
- D. 4

Answer: C

Explanation:

The state transition diagram provided shows three distinct states:

- ? Waiting for fingerprint
- ? Waiting for PIN
- ? Valid PIN/ask menu selection

Each state represents a different stage in the system's operation, with transitions based on user actions and system responses.

NEW QUESTION 28

Which of the following is a task the Author is responsible for, as part of a typical formal review?

- A. Determining the people who will be involved in the review
- B. Recording the anomalies found during the review meeting
- C. Identifying potential anomalies in the work product under review
- D. Fixing the anomalies found in the work product under review

Answer: C

Explanation:

This answer is correct because identifying potential anomalies in the work product under review is one of the tasks the Author is responsible for, as part of a typical formal review. The Author is the person who creates the work product to be reviewed, such as a requirement specification, a design document, or a test case. The Author's tasks include preparing the work product for the review, identifying potential anomalies in the work product, and fixing the anomalies found in the work product after the review. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.4.2.1

NEW QUESTION 33

Which of the following statements is true?

- A. A defect does not always produce a failure, while a bug always produces a failure
- B. A defect may cause a failure which, when occurring, always causes an error
- C. Failures can be caused by defects, but also by environmental conditions
- D. Bugs are defects found during component testing, while failures are defects found at higher test levels

Answer: C

Explanation:

Failures can be caused by defects, but also by environmental conditions. A failure is an event in which the software system does not perform a required function or performs a function incorrectly, according to the expected behavior. A defect is a flaw in the software system or a deviation from the requirements or the specifications, that may cause a failure. However, not all failures are caused by defects, as some failures may be caused by environmental conditions, such as hardware malfunctions, network interruptions, power outages, incompatible configurations, etc. Environmental conditions are factors that affect the operation of the software system, but are not part of the software system itself. The other statements are false, because:

? A defect does not always produce a failure, while a bug always produces a failure.

This statement is false, because a defect may or may not produce a failure, depending on the inputs, the outputs, the states, or the scenarios of the software system, and a bug is just another term for a defect, so it has the same possibility of producing a failure as a defect. For example, a defect in a rarely used feature or a hidden branch of the code may never produce a failure, while a defect in a frequently used feature or a critical path of the code may produce a failure often. A bug is not a different concept from a defect, but rather a synonym or a colloquial term for a defect, so it has the same definition and implications as a defect.

? A defect may cause a failure which, when occurring, always causes an error. This

statement is false, because an error is not a consequence of a failure, but rather a cause of a defect. An error is a human action or a mistake that produces a defect in the software system, such as a typo, a logic flaw, a requirement misunderstanding, etc. An error is not observable in the software system, but rather in the human mind or the human work products, such as the code, the design, the documentation, etc. A failure is not a cause of an error, but rather a result of a defect, which is a result of an error. For example, an error in the code may cause a defect in the software system, which may cause a failure in the software behavior.

? Bugs are defects found during component testing, while failures are defects found at higher test levels. This statement is false, because bugs and failures are not different types of defects, but rather different terms for defects and their manifestations. As mentioned before, bugs are just another word for defects, and failures are the events in which the software system does not perform as expected due to defects. Bugs and failures can be found at any test level, not only at component testing or higher test levels. Test levels are the stages of testing that correspond to the levels of integration of the software system, such as component testing, integration testing, system testing, and acceptance testing. Defects and failures can occur and be detected at any test level, depending on the test objectives, the test basis, the test techniques, and the test environment. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.1.2, Testing and Quality1

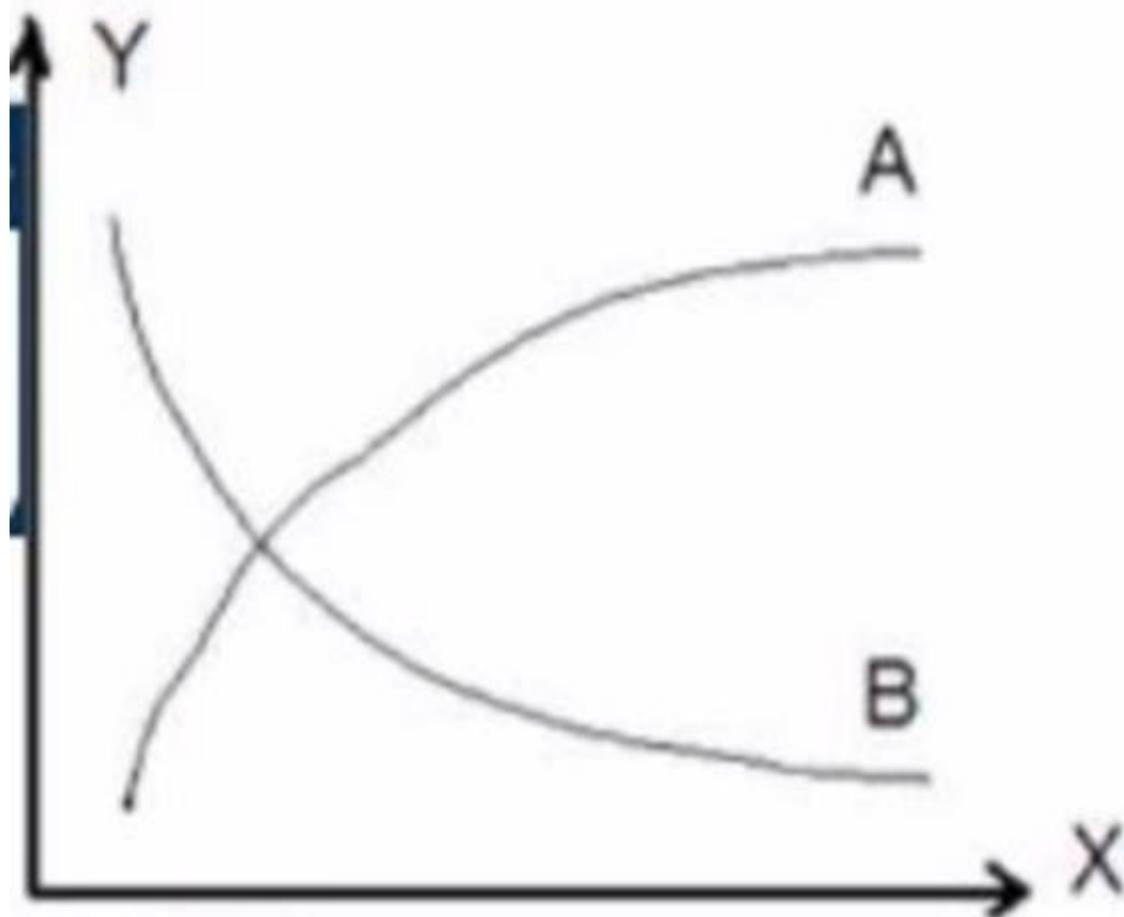
? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.2.1, Testing Principles1

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 1.3.1, Testing in Software Development Lifecycles1

? ISTQB® Glossary of Testing Terms v4.0, Failure, Defect, Bug, Environmental Condition, Error, Test Level2

NEW QUESTION 34

The following chart represents metrics related to testing of a project that was completed. Indicate what is represented by tie lines A, B and the axes X.Y



IL-MS-0218

A)

X - Time

Y - Cost

A - Cost of test (per week)

B - Cost of finding a single bug (per week)

B)

X - Time

Y - Number of defects

A - Number of open defects

B - Number of closed defects

C)

X - Time
 Y - Percent
 A - % of functional tests in the test suite
 B - % of non-functional tests in the test suite
 D)

X - Time
 Y - Count
 A - Total number of executed tests
 B - Number of open bugs

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

Explanation:

Option D correctly explains what is represented by the lines A, B and the axes X, Y in a testing metrics chart. According to option D:

- ? X-axis represents Time
- ? Y-axis represents Count
- ? Line A represents Number of open bugs
- ? Line B represents Total number of executed tests

This information is essential in understanding and analyzing the testing metrics of a completed project.

References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Section 2.5.1, Page 35.

NEW QUESTION 38

A company wants to reward each of its salespeople with an annual bonus that represents the sum of all the bonuses accumulated for every single sale made by that salesperson. The bonus for a single sale can take on the following four values: 3%, 5%, 7% and 10% (the percentage refers to the amount of the single sale). These values are determined on the basis of the type of customer (classified as "Basic" or "Premium") to which such sale was made, and on the amount of such sale classified into the following three groups G1, G2 and G3:

- [G1]: less than 300 euros
- [G2]: between 300 and 2000 euros
- [G3]: greater than 2000 euros

Which of the following is the minimum number of test cases needed to cover the full decision table associated with this scenario?

- A. 12
- B. 6
- C. 4
- D. 3

Answer: B

Explanation:

The minimum number of test cases needed to cover the full decision table associated with this scenario is 6. This is because the decision table has 4 conditions (type of customer and amount of sale) and 4 actions (bonus percentage). The conditions have 2 possible values each (Basic or Premium, and G1, G2 or G3), so the total number of combinations is $2 \times 2 \times 2 \times 2 = 16$. However, not all combinations are valid, as some of them are contradictory or impossible. For example, a sale cannot be both less than 300 euros and greater than 2000 euros at the same time. Therefore, we need to eliminate the invalid combinations and keep only the valid ones. The valid combinations are:

Type of customer	Amount of sale	Bonus percentage
Basic	G1	3%
Basic	G2	5%
Basic	G3	7%
Premium	G1	5%
Premium	G2	7%
Premium	G3	10%

These 6 combinations cover all the possible values of the conditions and actions, and they are the minimum number of test cases needed to cover the full decision table. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents,

NEW QUESTION 40

Which of the following statements about white-box testing is FALSE?

- A. Static testing can benefit from using code-related white-box test techniques during code reviews.
- B. White-box testing allows suggesting test cases for increasing coverage levels which are based on objective measures
- C. Achieving full code coverage for a component or a system ensures that it has been fully tested
- D. Black-box testing can benefit from using code-related white-box test techniques to increase confidence in the code.

Answer: C

Explanation:

Achieving full code coverage does not guarantee that the component or system is fully tested or free of defects. Code coverage metrics indicate the extent to

which the source code has been tested, but they do not account for the quality of the tests or whether all possible scenarios have been considered. Other types of testing, including functional, performance, and security testing, are necessary to ensure comprehensive testing. The ISTQB CTFL Syllabus v4.0 highlights that while high code coverage is beneficial, it does not equate to complete testing.

NEW QUESTION 45

A new web app aims at offering a rich user experience. As a functional tester, you have run some functional tests to verify that, before releasing the app, such app works correctly on several mobile devices, all of which are listed as supported devices within the requirements specification. These tests were performed on stable and isolated test environments where you were the only user interacting with the application. All tests passed, but in some of those tests you observed the following issue: on some mobile devices only, the response time for two web pages containing images was extremely slow. Based only on the given information, which of the following recommendation would you follow?

- A. You should open a defect report providing detailed information on which devices and by running which tests you observed the issue
- B. The issue is related to performance efficiency, not functionalit
- C. Thus, as a functional tester, you should not open any defect report as all the functional tests passed
- D. You should not open any defect report as the problem is most likely due to poor hardware equipment on the devices where you observed the issue
- E. You should not open any defect report and inform the test manager that the devices on which you observed the issue should no longer be supported so that they will be removed from the requirements specification

Answer: A

Explanation:

As a functional tester, you should open a defect report providing detailed information on which devices and by running which tests you observed the issue. A defect report is a document that records the occurrence, nature, and status of a defect detected during testing, and provides information for further investigation and resolution. A defect report should include relevant information such as the defect summary, the defectdescription, the defect severity, the defect priority, the defect status, the defect origin, the defect category, the defect reproduction steps, the defect screenshots, the defect attachments, etc. Opening a defect report is a good practice for any tester who finds a defect in the software system, regardless of the type or level of testing performed. The other options are not recommended, because:
? The issue is related to performance efficiency, not functionality, but that does not mean that as a functional tester, you should not open any defect report as all the functional tests passed. Performance efficiency is a quality characteristic that measures how well the software system performs its functions under stated conditions, such as the response time, the resource utilization, the throughput, etc. Performance efficiency is an important aspect of the user experience, especially for web applications that run on different devices and networks. Even if the functional tests passed, meaning that the software system met the functional requirements, the performance issue observed on some devices could still affect the user satisfaction, the usability, the reliability, and the security of the software system. Therefore, as a functional tester, you have the responsibility to report the performance issue as a defect, and provide as much information as possible to help the developers or the performance testers to investigate and resolve it.

NEW QUESTION 47

A company runs a pilot project for evaluation of a test automation tool. Which of the following is NOT a valid object of this pilot project?

- A. Get familiar with the functionality and options of the tool
- B. Check haw the tool fits to the existing test processes
- C. Train all testers on using the tool
- D. Decide upon standards for tool implementation

Answer: C

Explanation:

? A pilot project is a small-scale experiment or trial that is conducted to evaluate the feasibility, effectiveness, and suitability of a test automation tool before implementing it on a larger scale1.

? The objectives of a pilot project may vary depending on the context and scope of the test automation initiative, but some common ones are2:

? Therefore, option C is not a valid objective of a pilot project, as it is not necessary to train all testers on using the tool at this stage. Training all testers on using the tool would be more appropriate after the tool has been selected and approved for full-scale implementation, and after the standards and guidelines have been established. Training all testers on using the tool during the pilot project would be inefficient, costly, and premature, as the tool may not be suitable or effective for the intended purpose, or may be replaced by another tool later.

References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 82

? 2: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 83

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 84

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 85

NEW QUESTION 49

In which one of the following test techniques are test cases derived from the analysis of the software architecture?

- A. Black-box test techniques.
- B. Experience-based test techniques.
- C. Checklist-based test techniques.
- D. White-box test techniques.

Answer: D

Explanation:

White-box test techniques are test design techniques where the test cases are derived from the internal structure of the software, including its architecture, code, and logical flow. These techniques involve the tester having knowledge of the internal workings of the software to create test cases that ensure all possible paths and conditions are tested. This is in contrast to black-box test techniques, which focus on input-output behavior without considering the internal structure.Reference:ISTQB CTFL Syllabus V4.0, Section 4.3

NEW QUESTION 51

A possible risk of introducing test automation is:

- A. the tool may not be fit-for-purpose.

- B. the tool may create additional development dependencies.
- C. the tool may not be compatible with the development platform.
- D. the tool will be owned and maintained by developers and replace testers.

Answer: A

Explanation:

One possible risk of introducing test automation is that the selected tool may not be fit-for-purpose. This means that the tool might not meet the specific needs and requirements of the project, leading to inefficiencies and possibly failing to provide the expected benefits. It is crucial to evaluate and select the appropriate tool based on the project's context and objectives. The ISTQB CTFL syllabus highlights the importance of careful tool evaluation and selection to ensure it aligns with the testing goals and the development environment. References: ISTQB CTFL Syllabus, Section 6.2, "Potential Benefits and Risks of Test Automation."

NEW QUESTION 52

To be able to define testable acceptance criteria, specific topics need to be addressed. In the table below are the topics matched to an incorrect description. Match the topics (the left column) with the correct description (the right column)

- A. Mastered
- B. Not Mastered

Answer: A

NEW QUESTION 53

Mark the correct sentences:

- * Defects are a result of environmental conditions and are also referred to as "Failures"
- * A human mistake may produce a defect
- * A system may totally fail to operate correctly when a failure exists in it
- * When a defect exists in a system it may result in a failure
- * Defects occur only as a result of technology changes

- A. II, IV
- B. I, II
- C. IV, V
- D. II, III, IV

Answer: A

Explanation:

? The question is about marking the correct sentences among the given statements related to defects, failures, and mistakes. According to the ISTQB glossary, the definitions of these terms are:

? Therefore, out of the five given statements, only two are correct, namely:

? The other three statements are incorrect, namely: References:

? 1: ISTQB Glossary of Testing Terms 4.0, 2023, available at ISTQB and ASTQB).

NEW QUESTION 56

Which of the following statements is incorrect regarding the involvement of testers in the software development lifecycle (SDLC)?

- A. Testers should contribute to all activities in the SDLC and participate in design discussions.
- B. Testers should be involved from the beginning of the SDLC to increase understanding of design decisions and detect defects early.
- C. Testers should only be involved during the testing phase.
- D. Testers' involvement is essential; developers find it difficult to be objective.

Answer: C

Explanation:

Involving testers only during the testing phase is incorrect as per the ISTQB CTFL syllabus. Effective involvement of testers is crucial throughout the entire software development lifecycle (SDLC). This includes early stages such as requirement analysis and design, which allows testers to understand the design decisions and detect defects early. Early involvement helps in better understanding the project and ensures that quality is built into the product from the beginning. Furthermore, the ISTQB syllabus emphasizes the importance of testers contributing to all activities in the SDLC, including design discussions, to enhance defect detection and prevention.

References: ISTQB CTFL Syllabus, Section 2.1.1, "The Influence of Development Models on Testing" and Section 1.1.1, "Test Objectives."

NEW QUESTION 60

Following a risk-based testing approach you have designed 10 tests to cover a product risk with a high-risk level. You want to estimate, adopting the three-point test estimation technique, the test effort required to reduce the risk level to zero by executing those 10 tests. You made the following three initial estimates:

- most optimistic = 6 person hours
- most likely = 30 person hours
- most pessimistic = 54 person hours

Based only on the given information, which of the following answers about the three-point test estimation technique applied to this problem is true?

- A. The final estimate is between 22 person hours and 38 person hours
- B. The final estimate is exactly 30 person hours because the technique uses the initial most likely estimate as the final estimate
- C. The final estimate is between 6 person hours and 54 person hours
- D. The final estimate is exactly 30 person hours because the technique uses the arithmetic mean of the three initial estimates as the final estimate

Answer: A

Explanation:

The three-point test estimation technique is a method of estimating the test effort based on three initial estimates: the most optimistic, the most likely, and the most pessimistic. The technique uses a weighted average of these three estimates to calculate the final estimate, which is also known as the expected value. The

formula for the expected value is:

Expected value = (most optimistic + 4 * most likely + most pessimistic) / 6 Using the given values, the expected value is:

Expected value = (6 + 4 * 30 + 54) / 6 Expected value = 30 person hours

However, the expected value is not the only factor to consider when estimating the test effort. The technique also calculates the standard deviation, which is a measure of the variability or uncertainty of the estimates. The formula for the standard deviation is: Standard deviation = (most pessimistic - most optimistic) / 6

Using the given values, the standard deviation is:

Standard deviation = (54 - 6) / 6 Standard deviation = 8 person hours

The standard deviation can be used to determine a range of possible values for the test effort, based on a certain level of confidence. For example, using a 68% confidence level, the range is:

Expected value ?? standard deviation Using the calculated values, the range is: 30 ?? 8 person hours

Therefore, the final estimate is between 22 person hours and 38 person hours, which is option A.

References: ISTQB® Certified Tester Foundation Level Syllabus v4.01, Section 2.3.2, page 24-25; ISTQB® Glossary v4.02, page 33.

NEW QUESTION 64

Which of the following statements is NOT true about Configuration management and software testing?

- A. Configuration management helps maintain consistent versions of software artifacts.
- B. Configuration management supports the build process, which is essential for delivering a test release into the test environment.
- C. When testers report defects, they need to reference version-controlled items.
- D. Version controlled test ware increases the chances of finding defects in the software under test.

Answer: D

Explanation:

Reference:ISTQB CTFL Syllabus V4.0, Section 5.4

NEW QUESTION 65

Which of the following statements best describes the difference between product risk and project risk in software testing?

- A. Product risk refers to the risk associated with the project's schedule, budget, and resources, while project risk refers to the risk associated with the quality and functionality of the software product.
- B. Product risk refers to the risk associated with issues such as delays in work product deliveries, inaccurate estimates, while project risk refers to the risk associated with the project's schedule, budget, and resources.
- C. Product risk and project risk are essentially the same and can be used interchangeably.
- D. Product risk refers to the risk associated with delays in elements such as work product deliveries and inaccurate estimates, while project risk refers to the risk associated with issues such as user dissatisfaction.

Answer: B

Explanation:

Product risk involves the potential issues that can affect the quality and functionality of the software product, such as defects, performance problems, and usability issues. Project risk, on the other hand, relates to the risks that can impact the project's schedule, budget, and resources, such as delays, cost overruns, and resource constraints. Understanding both types of risks is crucial for managing and mitigating potential problems in software projects.

References:ISTQB CTFL Syllabus, Section 5.2.1, "Risk Management in Testing."

NEW QUESTION 66

From a testing perspective, configuration management

- A. Allows the expected results to be compared with the actual results.
- B. Allows the tracking of all changes to versions of the testware.
- C. Includes all activities that direct and control an organisation with regard to quality
- D. Focuses on configuring static analysis tools to choose the most suitable breadth and depth of analysis.

Answer: B

Explanation:

Configuration management in the context of testing involves the systematic control of changes to the configuration items, including testware such as test scripts, test data, and test environments. It ensures that all changes are tracked and recorded, enabling the version control and management of testware .

Option A is related to test execution rather than configuration management. Option C describes quality management in a broader sense, not specifically configuration management. Option D is specific to the configuration of tools, not the overall management of testware versions.

NEW QUESTION 68

You are an experienced tester on a project with incomplete requirements and under pressure to deploy.

What type of testing should you do?

- A. Decision-based testing.
- B. Checklist-based testing.
- C. Error guessing.
- D. Exploratory testing.

Answer: D

Explanation:

When working on a project with incomplete requirements and under pressure to deploy, exploratory testing is particularly suitable. This type of testing allows testers to use their expertise and intuition to explore the system's functionality and identify defects without needing detailed specifications. Exploratory testing is flexible and can quickly adapt to changes and gaps in the requirements.

NEW QUESTION 72

Which one of the following statements IS NOT a valid objective of testing?

- A. To build confidence in the level of quality of the test object.
- B. To find all defects in a product, ensuring the product is defect free.
- C. To find failures and defects
- D. To evaluate work products such as requirements, user stories, design, and code.

Answer: B

Explanation:

Reference:ISTQB CTFL Syllabus V4.0, Section 1.1.1

NEW QUESTION 75

The following decision table is used to assist a doctor in determining the drug therapy to prescribe for a patient (aged 6 to 65 years) diagnosed with acute sinusitis. The table consists of three Boolean conditions and six actions

	1	2	3	4	5	6	7	8
Conditions								
Is the patient over 18 years old?	F	F	F	F	T	T	T	T
Is the patient allergic to Penicillin?	F	F	T	T	F	F	T	T
Is the patient taking anticoagulant therapy?	F	T	F	T	F	T	F	T
Actions								
Amoxicillin is the therapy of choice					X			
Levofloxacin is the therapy of choice			X				X	
Cefuroxime is the therapy of choice	X							
Necessary consultation with the hematologist		X		X		X		X
Full dosage recommended for 10 days					X		X	
Half of the full recommended dosage for 10 days	X		X					

Based only on the given information, which of the following statements is TRUE?

- A. Column 7 represents an impossible situation and thus can be deleted
- B. Columns 1 and 3 can be merged into a single column
- C. Columns 2, 4, 6 and 8 can be merged into a single column
- D. Columns 5 and 7 can be merged into a single column

Answer: B

Explanation:

Decision tables are used to model complex decision logic by considering different combinations of conditions and actions. Based on the given decision table for prescribing drug therapy:

? Column 1 and Column 3 both result in the same actions (prescribing Amoxicillin).

? These columns can be merged because the actions taken do not depend on

whether the patient is taking anticoagulant therapy (both are 'T' for this condition). Thus, combining these columns simplifies the decision table without losing any information. Reference: ISTQB CTFL Syllabus V4.0, Chapter 4.2.3, Decision Table Testing.

NEW QUESTION 76

Can "cost" be regarded as Exit criteria?

- A. Ye

- B. Spending too much money on testing will result in an unprofitable product, and having cost as an exit criterion helps avoid this
- C. No
- D. The financial value of product quality cannot be estimated, so it is incorrect to use cost as an exit criterion
- E. Yes
- F. Going by cost as an exit criterion constrains the testing project which will help achieve the desired quality level defined for the project
- G. No The cost of testing cannot be measured effectively, so it is incorrect to use cost as an exit criterion

Answer: A

Explanation:

Cost can be regarded as an exit criterion for testing, because it is a factor that affects the profitability and feasibility of the software product. Testing is an investment that aims to improve the quality and reliability of the software product, but it also consumes resources, such as time, money, and human effort. Therefore, testing should be planned and executed in a way that balances the cost and benefit of testing activities. Having cost as an exit criterion helps to avoid spending too much money on testing, which may result in an unprofitable product or a loss of competitive advantage. Cost can also help to prioritize and focus the testing efforts on the most critical and valuable features and functions of the software product. However, cost should not be the only exit criterion for testing, as it may not reflect the true quality and risk level of the software product. Other exit criteria, such as defect rate, test coverage, user satisfaction, etc., should also be considered and defined in the test plan.

The other options are incorrect, because they either deny the importance of cost as an exit criterion, or they make false or unrealistic assumptions about the cost of testing. Option B is incorrect, because the financial value of product quality can be estimated, for example, by using cost-benefit analysis, return on investment, or cost of quality models. Option C is incorrect, because going by cost as an exit criterion does not necessarily constrain the testing project or help achieve the desired quality level. Cost is a relative and variable factor that depends on the scope, complexity, and context of the software product and the testing project. Option D is incorrect, because the cost of testing can be measured effectively, for example, by using metrics, such as test effort, test resources, test tools, test environment, etc.

NEW QUESTION 80

Which of the following statements about error guessing is true?

- A. Error guessing is a system that adopts artificial intelligence to predict whether software components are likely to contain defects or not
- B. Experienced testers, when applying error guessing, rely on the use of a high-level list of what needs to be tested as a guide to find defects
- C. Error guessing refers to the ability of a system or component to continue normal operation despite the presence of erroneous inputs
- D. Experienced testers, when applying error guessing technique, can anticipate where errors, defects and failures have occurred and target their tests at those issues

Answer: D

Explanation:

This answer is correct because error guessing is a test design technique where the experience and intuition of the tester are used to anticipate where errors, defects and failures have occurred or are likely to occur, and to design test cases to expose them. Error guessing can be based on factors such as the complexity of the system or component, the known or suspected weaknesses of the system or component, the previous history of defects, or the common types of errors in the domain or technology. Error guessing can be used as a complementary technique to other more systematic or formal techniques, or when there is insufficient information or time to apply them. References: ISTQB Glossary of Testing Terms v4.0, ISTQB Foundation Level Syllabus v4.0, Section 2.3.2.5

NEW QUESTION 82

Which of the following statements describes regression testing?

- A. Retesting of a fixed defect
- B. Testing of an already tested program
- C. Testing of new functionality in a program
- D. Regression testing applies only to functional testing
- E. Tests that do not have to be repeatable, because they are only used once
- F. II, IV, V
- G. I, III, IV
- H. II
- I. I, IV

Answer: C

Explanation:

Regression testing is the re-running of functional and non-functional tests to ensure that previously developed and tested software still performs as expected after a change. It does not involve retesting of a fixed defect, testing of new functionality, or applying only to functional testing. Tests that are used for regression testing should be repeatable, because they are used to verify the stability of the software after each change. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 4, Section 4.2.2, Page 291; ISTQB Glossary of Testing Terms v4.0, Page 292

NEW QUESTION 86

An alphanumeric password must be between 4 and 7 characters long and must contain at least one numeric character, one capital (uppercase) letter and one lowercase letter of the alphabet.

Which one of the following sets of test cases represents the correct outcome of a two-value boundary value analysis applied to the password length? (Note: test cases are separated by a semicolon)

- A. 1xA;aB11;Pq1ZZab;7iDD0a1x
- B. aB11;99rSp:5NnN10;7iDD0a1x
- C. 1xB: aB11: 99rSp: 5NnN10; 4NnN10T; 44ghWn19
- D. 1RhT;rSp53;3N3e10;8sBdby

Answer: D

Explanation:

The correct outcome of a two-value boundary value analysis applied to the password length is the set of test cases represented by option D. Boundary value analysis is a test design technique that focuses on the values at the boundaries of an equivalence partition, such as the minimum and maximum values, or the values just above and below the boundaries. A two-value boundary value analysis uses two values for each boundary, one representing the valid value and one representing the invalid value. For example, if the valid range of values is from 4 to 7, then the two values for the lower boundary are 3 and 4, and the two values

for the upper boundary are 7 and 8. The test cases in option D use these values for the password length, while also satisfying the other requirements of the password, such as containing at least one numeric character, one capital letter, and one lowercase letter. The test cases in option D are:

? 1RhT: a 4-character password that is valid

? rSp53: a 5-character password that is valid

? 3N3e10: a 6-character password that is valid

? 8sBdby: an 8-character password that is invalid The test cases in the other options are incorrect, because they either use values that are not at the boundaries of the password length, or they do not meet the other requirements of the password. For example, the test cases in option A are:

? 1xA: a 3-character password that is invalid, but it does not contain a capital letter

? aB11: a 4-character password that is valid

? Pq1ZZab: a 7-character password that is valid

? 7iDD0a1x: an 8-character password that is invalid References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 sources and documents:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0, Chapter 2.2.1, Black-box Test Design Techniques1

? ISTQB® Glossary of Testing Terms v4.0, Boundary Value Analysis, Equivalence Partition2

NEW QUESTION 91

Which of the following statements about the shift-left approach is FALSE?

- A. The shift-left approach can only be implemented with test automation
- B. The shift-left approach in testing is compatible with DevOps practices.
- C. The shift-left approach can involve security vulnerabilities
- D. The shift-left approach can be supported by static analysis tools.

Answer: A

Explanation:

In a formal review process, the recorder's role is typically responsible for documenting the findings of the review team, including action items, decisions, and recommendations. This ensures that there is an accurate record of what was discussed and agreed upon, facilitating follow-up and continuous improvement. Therefore, statement C is correct as per the ISTQB CTFL syllabus.

NEW QUESTION 94

.....

Relate Links

100% Pass Your CTFL4 Exam with ExamBible Prep Materials

<https://www.exambible.com/CTFL4-exam/>

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - <https://www.exambible.com/>