

ISTQB

Exam Questions ISTQB-CTFL

ISTQB-Foundation Level Exam



NEW QUESTION 1

Consider the following testing levels:

- 1) Component Testing
- 2) Integration Testing
- 3) System Testing
- 4) Acceptance Testing

Which of the following statements is true?

- A. Integration and system testing are applicable when V-model is used. Component and acceptance testing are applicable when iterative development models are used.
- B. All the testing levels are applicable to V-model for software development.
- C. Only acceptance testing is applicable for iterative models.
- D. Acceptance testing is applicable for all software development model.
- E. Component and system testing are applicable only for the V-model.
- F. All testing levels are applicable, independent of which software development life-cycle process (V-model, iterative, incremental) is used.
- G. iterative, incremental) is used.

Answer: D

Explanation:

All testing levels are applicable, independent of which software development life-cycle process (V-model, iterative, incremental) is used. Testing levels are defined based on the scope and objectives of testing, not on the software development model. Component testing, integration testing, system testing and acceptance testing are common testing levels that can be applied to any software development model, as long as they are planned and executed properly. The V-model is a software development model that emphasizes the relationship between each development phase and its corresponding testing phase. Iterative and incremental models are software development models that divide the development process into smaller cycles or iterations, where each iteration produces a working version of the software that can be tested and evaluated. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 18.

NEW QUESTION 2

Which of the following statements regarding inspection is NOT true?

- A. An inspection may be led by a trained moderator who shall not be the author.
- B. The main purpose of an inspection is to find solutions to the problems.
- C. An inspection can be performed by peers.
- D. An inspection shall follow a formal process based on rules and checklists with entry and exit criteria.

Answer: B

Explanation:

An inspection is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. An inspection can be performed by peers with different roles, such as moderator, author, reviewer and scribe. The following statement about inspection is not true:

? B) The main purpose of an inspection is to find solutions to the problems. This statement is not true, as the main purpose of an inspection is to find defects or issues in a work product, not to find solutions to the problems. Finding solutions to the problems is a debugging or problem-solving activity that is usually performed by the author or developer after receiving the inspection report. The following statements about inspection are true:

? A) An inspection may be led by a trained moderator who shall not be the author.

This statement is true, as an inspection requires a moderator role who leads the inspection process and ensures that it follows the rules and standards. The moderator should be trained in inspection techniques and should not be the author of the work product under inspection, in order to avoid bias or conflict of interest.

? C) An inspection can be performed by peers. This statement is true, as an inspection involves peer review, which means that the work product under inspection is evaluated by people who have similar roles or expertise as the author, but who are not directly involved in creating or modifying the work product.

? D) An inspection shall follow a formal process based on rules and checklists with entry and exit criteria. This statement is true, as an inspection follows a formal process that consists of six main steps: planning, kick-off meeting, individual preparation, review meeting, rework and follow-up. Each step has defined rules and checklists to guide the participants and ensure consistency and quality. Each step also has entry and exit criteria to ensure that the prerequisites and objectives are met before moving to the next step. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 28-29.

NEW QUESTION 3

Who of the following has the best knowledge to decide what tests in a test project should be automated?

- A. The developer
- B. The customer
- C. The development manager
- D. The test leader

Answer: D

Explanation:

The test leader is the person who is responsible for planning, monitoring, and controlling the test activities and resources in a test project. The test leader should have the best knowledge of the test objectives, scope, risks, resources, schedule, and quality criteria. The test leader should also be aware of the test automation criteria, such as the execution frequency, the test support, the team education, the roles and responsibilities, and the devs and testers collaboration¹. Based on these factors, the test leader can decide which tests are suitable for automation and which are not, and prioritize them accordingly. The test leader can also coordinate with the test automation engineers, the developers, and the stakeholders to ensure the alignment of the test automation strategy with the test project goals and expectations. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 2, Section 2.3.1, Page 152; ISTQB Glossary of Testing Terms v4.0, Page 403; ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 6, Section 6.1.1, Page 514; Top 8 Test Automation Criteria You Need To Fulfill - QAMIND1

NEW QUESTION 4

Which of the following is the most important task of a typical test leader?

- A. To automate tests.

- B. To prepare and acquire test data.
- C. To set up the test environment.
- D. To coordinate the test strategy with project managers.

Answer: D

Explanation:

The most important task of a typical test leader is to coordinate the test strategy with project managers. The test strategy is a high-level document that defines the general approach and objectives of testing for a project or an organization. The test leader is responsible for defining, documenting, communicating, and implementing the test strategy in alignment with the project goals and constraints. The test leader also needs to coordinate with project managers and other stakeholders to ensure that the test strategy is feasible, effective, and efficient. The other options are not the most important tasks of a typical test leader. To automate tests is a task of a test automation engineer or a test automation specialist. To prepare and acquire test data is a task of a test analyst or a test engineer. To set up the test environment is a task of a test environment manager or a test environment specialist. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 13.

NEW QUESTION 5

Given the following statements:

- * 1. It can prevent defects by manual examination of the functional specification
- * 2. It is effective since it can be performed very early in the software development life cycle
- * 3. It can detect the failures in the running application
- * 4. It can help eliminate defects in user stories
- 5. It can verify externally visible behaviors

Which set of statements represent values ONLY for static testing?

- A. 1,3, 4,5
- B. 2,4,5
- C. 1,2,4
- D. 1,2, 3, 4,5

Answer: C

Explanation:

Static testing involves reviewing and inspecting the code, requirements, or design documents without executing the code. It can prevent defects, is effective early in the software development life cycle, and can help eliminate defects in user stories. Option 1: "It can prevent defects by manual examination of the functional specification" - This is a value of static testing.

Option 2: "It is effective since it can be performed very early in the software development life cycle" - This is a value of static testing.

Option 3: "It can detect the failures in the running application" - This is a value of dynamic testing, not static testing.

Option 4: "It can help eliminate defects in user stories" - This is a value of static testing. Option 5: "It can verify externally visible behaviors" - This is a value of dynamic testing, not static testing.

Therefore, the correct set of statements representing values only for static testing is 1, 2, 4, which corresponds to answer C.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 6

Which of the following coverage criteria results in the highest coverage for state transition based test cases?

- A. Can't be determined
- B. Covering all transitions at least once
- C. Covering only start and end states
- D. Covering all states at least once

Answer: B

Explanation:

Covering all transitions at least once is the highest coverage criterion for state transition based test cases, because it ensures that every possible change of state is tested at least once. This means that all the events that trigger the transitions, as well as the actions and outputs that result from the transitions, are verified.

Covering all transitions at least once also implies covering all states at least once, but not vice versa. Therefore, option D is not the highest coverage criterion.

Option C is the lowest coverage criterion, because it only tests the initial and final states of the system or component, without checking the intermediate states or transitions. Option A is incorrect, because the coverage criteria for state transition based test cases can be determined and compared based on the number of transitions and states covered. References = CTFL 4.0 Syllabus, Section 4.2.3, page 49-50.

NEW QUESTION 7

Which of the following BEST describes a benefit of test automation?

- A. More subjective assessment
- B. Reduction in repetitive manual work
- C. Availability of the test automation tool vendor
- D. Negligible effort to maintain the test assets generated by the tool

Answer: B

Explanation:

Test automation provides numerous benefits to software testing, and one of the key advantages is the reduction of repetitive manual work. This benefit is explicitly covered in the ISTQB Foundation Level Syllabus (v4.0).

Test automation allows testers to automate repetitive tasks such as regression testing, freeing up their time to focus on more complex and exploratory testing. This leads to improved efficiency and helps in avoiding human errors associated with repetitive tasks. Option A: "More subjective assessment" contradicts the benefit of automation as it focuses on objectivity.

Option C: "Availability of the test automation tool vendor" is not a direct benefit of test automation, although vendor support can be valuable.

Option D: "Negligible effort to maintain the test assets" is misleading as maintaining automated tests often requires effort and attention to changes in the system under test. Therefore, the correct answer is B (ISTQB not-for-profit association) (ISTQB). References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 8

Which of the following applications will be the MOST suitable for testing by Use Cases

- A. Accuracy and usability of a new Navigation system compared with previous system
- B. A billing system used to calculate monthly charge based on large number of subscribers parameters
- C. The ability of an Anti virus package to detect and quarantine a new threat
- D. Suitability and performance of a Multi media (audio video based) system to a new operating system

Answer: A

Explanation:

A new navigation system compared with a previous system is the most suitable application for testing by use cases, because it involves a high level of interaction between the user and the system, and the expected behavior and outcomes of the system are based on the user's needs and goals. Use cases can help to specify the functional requirements of the new navigation system, such as the ability to enter a destination, select a route, follow the directions, receive alerts, etc. Use cases can also help to compare the accuracy and usability of the new system with the previous system, by defining the success and failure scenarios, the preconditions and postconditions, and the alternative flows of each use case. Use cases can also help to design and execute test cases that cover the main and exceptional paths of each use case, and to verify the satisfaction of the user's expectations.

The other options are not the most suitable applications for testing by use cases, because they do not involve a high level of interaction between the user and the system, or the expected behavior and outcomes of the system are not based on the user's needs and goals. A billing system used to calculate monthly charge based on a large number of subscriber parameters is more suitable for testing by data-driven testing, which is a technique for testing the functionality and performance of a system or component by using a large set of input and output data. The ability of an antivirus package to detect and quarantine a new threat is more suitable for testing by exploratory testing, which is a technique for testing the functionality and security of a system or component by using an informal and flexible approach, based on the tester's experience and intuition. The suitability and performance of a multimedia (audio video based) system to a new operating system is more suitable for testing by compatibility testing, which is a technique for testing the functionality and performance of a system or component by using different hardware, software, or network environments. References = CTFL 4.0 Syllabus, Section 3.1.1, page 28-29; Section 4.1.1, page 44-45; Section 4.2.1, page 47-48.

NEW QUESTION 9

The following requirement is given "Set X to be the sum of Y and Z". All the following four implementations have bugs.

Which one of the following bugs can be caught by Static Analysis?

- A. `int x = 1. int y = 2. int y = 3.X = y=z;`
- B. `int x = 1. int y = 2. int z = 3.X = z-y`
- C. `int x = 1. int y = 2. int z = 3.Z = x +y`
- D. `int y = 2 int z = 3. Y = z+y`

Answer: A

Explanation:

Static analysis is a technique that analyzes the source code or other software artifacts without executing them. Static analysis can detect defects such as syntax errors, coding standards violations, potential security vulnerabilities, or logical flaws. Static analysis can catch the bug in the first implementation, as it contains two syntax errors: the variable y is declared twice, and the assignment statement `X = y=z` is invalid. Static analysis cannot catch the bugs in the other three implementations, as they are logical errors that do not violate any syntax rules, but produce incorrect results. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 25-26.

NEW QUESTION 10

Which of the following software development models BEST exemplifies a model that does NOT support the principle of early testing?

- A. The iterative development model
- B. The V-model
- C. The Waterfall model
- D. The incremental development model

Answer: C

Explanation:

The Waterfall model exemplifies a software development model that does not support the principle of early testing. In the Waterfall model, each phase must be completed before the next begins, which delays testing until after the completion of the earlier phases like requirements gathering and design. This can often result in finding defects later in the development cycle, making them more expensive and time-consuming to fix (ISTQB not-for-profit association) (ISTQB not-for-profit association).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf

? ISTQB News Release on CTFL v4.0: <https://www.istqb.org/news/posts/istqb-releases-certified-tester-foundation-level-v40-ctfl/>

NEW QUESTION 10

Which of the following would be a key difference between a peer review of code and static analysis of code using a tool?

- A. A peer review finds defects while static analysis finds failures.
- B. Static analysis targets the code technically whereas Peer review is applicable to further aspects.
- C. Peer reviews cannot find missing requirements whereas static analysis can
- D. A peer review finds failures while static analysis finds defects.

Answer: B

Explanation:

The key difference between a peer review of code and static analysis of code using a tool lies in their approaches and scope. A peer review is a manual inspection of the code by peers or colleagues, focusing not only on the technical aspects of the code but also on other elements such as design, compliance with

standards, and maintainability. Peer reviews can identify defects, suggest improvements, and ensure that the code adheres to best practices and team standards. On the other hand, static analysis is an automated process performed by tools designed to analyze the code without executing it. These tools can detect potential issues such as syntax errors, vulnerabilities, and code smells based on predefined rules and patterns. While static analysis is technically focused, it lacks the broader perspective that human reviewers can provide, such as evaluating the code's maintainability or adherence to project-specific standards. Therefore, static analysis targets the code technically, whereas peer review encompasses a wider range of aspects, making option B the correct answer.

NEW QUESTION 11

Which of the following definitions is NOT true?

- A. Test data preparation tools fill databases, create files or data transmissions to set up test data to be used during the execution of tests.
- B. Test execution tools execute test objects using automated test scripts.
- C. Test Management tools monitor and report on how a system behaves during the testing activities.
- D. Test comparators determine differences between files, databases or test results.

Answer: C

Explanation:

Test Management tools are designed to support the planning, execution, and monitoring of the testing process. They provide features for managing test cases, test runs, tracking defects, and reporting on testing activities. However, the statement in option C describes Test Management tools as monitoring and reporting on the system's behavior during testing activities, which is not accurate. Test Management tools focus on the testing process itself rather than on the behavior of the system under test.

? Test data preparation tools (A) indeed create and manage test data for use during test execution.

? Test execution tools (B) automate the execution of test cases and the comparison of actual outcomes against expected results.

? Test comparators (D) are tools that compare actual outcomes with expected outcomes, highlighting discrepancies.

Therefore, option C is the correct answer as it inaccurately describes the function of Test Management tools.

NEW QUESTION 16

Out of the following, what is not needed to specify in defect report?

- A. Test environment details
- B. How to reproduce the defect
- C. How to fix the defect
- D. Severity and priority

Answer: C

Explanation:

A defect report is a document that records the details of a defect found during testing. A defect report typically contains the following items:

? Identifier: A unique identifier for the defect report

? Summary: A concise summary of the defect

? Description: A detailed description of the defect, including the steps to reproduce it, the expected and actual results, and any relevant screenshots or logs

? Severity: The degree of impact that the defect has on the system

? Priority: The level of urgency for resolving the defect

? Status: The current state of the defect, such as new, open, resolved, closed, etc.

? Resolution: The action taken to resolve the defect, such as fix, workaround, reject, etc. Out of these items, the one that is not needed to specify in a defect report is how to fix the defect. How to fix the defect is a technical solution that is usually determined by the developer who is assigned to resolve the defect. How to fix the defect is not part of the defect report, but rather part of the code change or patch that is delivered to fix the defect. The other items are needed to specify in a defect report, as they provide essential information for identifying, tracking and resolving defects. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 32-33.

NEW QUESTION 19

Which of the following is a typical product risk?

- A. Poor usability of the software
- B. A problem in the code developed by a 3rd party
- C. Low quality of the configuration data, test data and tests
- D. Problem in defining the right requirements

Answer: A

Explanation:

A typical product risk involves issues directly related to the software product's functionality, performance, usability, reliability, etc. Option A, "Poor usability of the software," directly impacts the end-user's interaction with the software and is a quality attribute of the product itself, making it a product risk. Options B, "A problem in the code developed by a 3rd party," C, "Low quality of the configuration data, test data and tests," and D, "Problem in defining the right requirements," can be considered either product or project risks depending on the context, but option A is the most directly associated with a typical product risk concerning the quality and usability of the software.

NEW QUESTION 23

Which of the following is the BEST reason for selecting a particular type of software development lifecycle model?

- A. The project manager's preference
- B. Tester skill level with the software development lifecycle model
- C. The project team's overall familiarity with the model
- D. The type of product being developed

Answer: D

Explanation:

The choice of a software development lifecycle (SDLC) model is primarily influenced by the type of product being developed. Different products and project

requirements may demand different SDLC models to address specific challenges and needs efficiently. For instance, a complex, safety-critical product might best be served by a Waterfall model due to its structured nature and phase dependencies, while a more iterative and incremental model might be suited for projects requiring frequent feedback and changes. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 2.1 "Software Development Lifecycles".

NEW QUESTION 24

Which of the following project scenario gives the BEST example where maintenance testing should be triggered?

- A. Completion of architecture of the bank system
- B. Release of the early draft of the low level project design of an IoT application
- C. Defect was found in a pre-released version of the customer service application
- D. Delivery of the hot fix to mobile operating system and ensuring that it still works

Answer: D

Explanation:

Maintenance testing is triggered by changes such as bug fixes, enhancements, or environmental changes.

Option A: "Completion of architecture of the bank system" is not a typical scenario for maintenance testing, as it describes a design phase rather than an operational change. Option B: "Release of the early draft of the low level project design of an IoT application" is again not suitable for maintenance testing, as it refers to the design phase.

Option C: "Defect was found in a pre-released version of the customer service application" is closer but not quite accurate, as maintenance testing focuses on changes mad (ISTQB not-for-profit association)system is released.

Option D: "Delivery of the hot fix to mobile operating system and ensuring that it still works" is the best example as it directly involves testing after a fix has been implemented. Therefore, the correct answer is D6†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 25

Why is it important to select a test technique?

- A. There are usually loo many test cases that may be run on a syste
- B. Test techniques help reduce the number of tests.
- C. The only way to test a software application is by using well proven test techniques.
- D. Selecting the right test technique in a given situation Increases the effectiveness of the test process Oy creating tests with higher chance of finding bugs.
- E. Test techniques define the number of regression cycles, which in turn impact the project schedule.

Answer: C

Explanation:

Selecting the right test technique is crucial because different techniques are suited to different types of testing and can significantly increase the effectiveness of the testing process by creating tests that are more likely to find defects. While reducing the number of tests (A) and defining the number of regression cycles (D) are considerations in the testing process, they are not the primary reasons for selecting a test technique. The assertion that the only way to test a software application is by using well-proven test techniques (B) is too restrictive and does not acknowledge the adaptability required in testing to suit different contexts and objectives. Therefore, option C is the most comprehensive reason, as it focuses on the effectiveness and efficiency of testing, leading to the creation of high-quality tests that have a higher chance of finding bugs.

NEW QUESTION 30

Why should you choose a test technique?

- A. Because you need to match the way you test to the content of the product under test
- B. Because of the time constraints that usually accompany a test project
- C. Because this way you cover the full scope of the product's functionality
- D. Because choosing a test technique is a common practice in software testing

Answer: A

Explanation:

You should choose a test technique because you need to match the way you test to the content of the product under test. A test technique is a method or process for deriving and selecting test cases based on some criteria or rules. Different test techniques are suitable for different types of software products, depending on their characteristics, functionalities, requirements, specifications, risks, etc. Choosing a test technique helps to ensure that the test cases are relevant, effective, and efficient for the product under test. The other options are not correct reasons to choose a test technique. Time constraints are not a factor for choosing a test technique, but rather for prioritizing or optimizing testing activities. Covering the full scope of the product's functionality is not a guarantee of choosing a test technique, but rather a goal of testing. Choosing a test technique is not a common practice in software testing, but rather a professional skill and responsibility.

Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 31.

NEW QUESTION 35

Consider the following code

```
int premium=2500; if (age<30)
{
    premium = premium +1500;
}
```

Which options suits for a correct combination of Boundary value and expected result. Assume first number as boundary followed by expected result.

- A. * 29, 1500* 30. 2500
- B. * 29. 4000* 30. 2500
- C. * 29, 2500* 30. 1500
- D. * 30, 1500* 31,2500

Answer: B

Explanation:

In the given code snippet, the premium is increased by 1500 if the age is less than 30. Therefore, at the boundary value of age 29, the premium should be 2500 + 1500 = 4000, and at age 30, the premium should remain at its initial value of 2500, as the condition is no longer met. Option B correctly reflects this with 29, 4000 and 30, 2500 as the boundary value and the expected results, respectively.

NEW QUESTION 38

While repotting a defect, which attribute indicates the degree of impact that the defect has on the system?

- A. Priority
- B. Severity
- C. Status
- D. Description

Answer: B

Explanation:

In defect reporting, the attribute that indicates the degree of impact that the defect has on the system is the severity. Severity reflects the seriousness of the defect in terms of its impact on the operation of the system, ranging from minor issues that do not significantly affect the system's functionality to critical defects that can cause system failure. Therefore, option B is the correct answer.

NEW QUESTION 43

Which of the following can be considered a VALID exit criterion? I Estimates of defect density or reliability measures.

II. The completion and publication of an exhaustive Test Report.

III. Accuracy measures, such as code, functionality or risk coverage. IV Residual risks such as lack of code coverage in certain areas.

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

Answer: A

Explanation:

An exit criterion is a condition that defines when a test activity has been completed or when a test phase can be concluded. An exit criterion can be based on various factors, such as:

? I) Estimates of defect density or reliability measures. These are quantitative measures that indicate how many defects are present in the software product or how likely it is to fail under certain conditions. These can be used as exit criteria to ensure that the software product meets a certain level of quality or performance before moving to the next phase or releasing it to the customer.

? III) Accuracy measures, such as code coverage, functionality coverage or risk coverage. These are quantitative measures that indicate how much of the software product has been tested in terms of its code, functionality or risk. These can be used as exit criteria to ensure that the test suite is adequate or complete before moving to the next phase or releasing it to the customer.

? IV) Residual risks, such as lack of code coverage in certain areas, unresolved defects or unknown factors. These are qualitative measures that indicate the remaining risks or uncertainties associated with the software product after testing. These can be used as exit criteria to ensure that the residual risks are acceptable or manageable before moving to the next phase or releasing it to the customer. The following factor is not a valid exit criterion:

? II) The completion and publication of an exhaustive Test Report. This is not a valid exit criterion, as it does not reflect the quality or completeness of the testing process or product. A test report is a document that summarizes the results and outcomes of a test activity or phase. A test report can be used as an input for deciding whether to exit a test activity or phase, but it is not a condition that defines when to exit. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 13; Chapter 6, page 58-59.

NEW QUESTION 44

Which test approach will best fit a new project, with little documentation and high probability for bugs?

- A. Exploratory testing
- B. Requirements based testing
- C. Metric based approach
- D. Regression testing

Answer: A

Explanation:

Exploratory testing is an approach to testing that emphasizes learning, test design and test execution at the same time. Exploratory testing relies on the tester's skills, creativity and intuition to explore the software under test and discover defects. Exploratory testing is suitable for a new project with little documentation and high probability for bugs, as it can help uncover unknown requirements, assumptions and risks. Exploratory testing is not requirements based testing, which is an approach to testing that derives test cases from documented requirements or specifications. Requirements based testing is not feasible for a new project with little documentation, as it requires clear and complete requirements to be available. Exploratory testing is not metric based approach, which is an approach to testing that uses quantitative measures to monitor and control the testing process and evaluate the quality of the software product. Metric based approach is not effective for a new project with high probability for bugs, as it may not capture all aspects of quality and may lead to false confidence or unrealistic expectations. Exploratory testing is not regression testing, which is an approach to testing that verifies that previously tested software still performs correctly after changes. Regression testing is not relevant for a new project with no previous versions or baselines. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 5, page 47-48.

NEW QUESTION 45

In maintenance testing, what is the relationship between impact analysis and regression testing?

- A. Impact analysis requires a regression testing for only the tests that have detected faults in previous SW release
- B. There is no relationship between impact analysis and regression testing.
- C. Impact analysis requires a regression testing for all program elements which were newly integrated (new functionalities).
- D. The impact analysis is used to evaluate the amount of regression testing to be performed.

Answer: D

Explanation:

In maintenance testing, the relationship between impact analysis and regression testing is that the impact analysis is used to evaluate the amount of regression testing to be performed. Maintenance testing is a type of testing that is performed on an existing software product after it has been delivered or deployed, in order to ensure that it still meets its requirements and functions correctly after a change or a modification. Maintenance testing can be triggered by various reasons, such as corrective maintenance (fixing defects), adaptive maintenance (adapting to new environments), perfective maintenance (improving performance), preventive maintenance (avoiding future problems), etc. Impact analysis is a technique that is used to assess the extent and nature of changes introduced by maintenance activities on the software product or project. Impact analysis helps to identify which parts of the software product are affected by the changes, which parts need to be modified or updated accordingly, which parts need to be retested or verified for correctness or compatibility, etc. Regression testing is a type of testing that verifies that previously tested software still performs correctly after a change or a modification. Regression testing helps to detect any side effects or unintended consequences of maintenance activities on the software product's functionality or quality. Regression testing can be performed at various levels and scopes depending on the impact analysis results. Therefore, in maintenance testing, impact analysis is used to evaluate the amount of regression testing to be performed. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 20.

NEW QUESTION 48

A software module to be used in a mission critical application incorporates an algorithm for secure transmission of data. Which review type is most appropriate to ensure high quality and technical correctness of the algorithm?

- A. Walkthrough
- B. Informal Review
- C. Technical Review
- D. Management Review

Answer: C

Explanation:

A technical review is a type of formal review that involves a team of technical experts who evaluate a software product against a set of predefined quality criteria. A technical review is suitable for ensuring high quality and technical correctness of complex or critical software components, such as algorithms, architectures or designs. A technical review is not a walkthrough, which is an informal review led by the author of the work product. A technical review is not an informal review, which is a review that does not follow a defined process and has no formal entry or exit criteria. A technical review is not a management review, which is a type of formal review that focuses on business aspects and project progress. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 29-30.

NEW QUESTION 53

Which of the following issues cannot be identified by static analysis tools?

- A. Very low MTBF (Mean Time Between failure)
- B. Potentially endless loops
- C. Referencing a variable with an undefined value
- D. Security vulnerabilities

Answer: A

Explanation:

Static analysis tools are software tools that examine the source code of a program without executing it. They can detect various types of issues, such as syntax errors, coding standards violations, security vulnerabilities, and potential bugs¹². However, static analysis tools cannot identify issues that depend on the runtime behavior or performance of the program, such as very low MTBF (Mean Time Between failure)³. MTBF is a measure of the reliability of a system or component. It is calculated by dividing the total operating time by the number of failures. MTBF reflects how often a system or component fails during its expected lifetime. Static analysis tools cannot measure MTBF because they do not run the program or observe its failures. MTBF can only be estimated by dynamic testing, which involves executing the program under various conditions and collecting data on its failures⁴. Therefore, very low MTBF is an issue that cannot be identified by static analysis tools. The other options, such as potentially endless loops, referencing a variable with an undefined value, and security vulnerabilities, are issues that can be identified by static analysis tools. Static analysis tools can detect potentially endless loops by analyzing the control flow and data flow of the program and checking for conditions that may never become false⁵. Static analysis tools can detect referencing a variable with an undefined value by checking the scope and initialization of variables and reporting any use of uninitialized variables⁶. Static analysis tools can detect security vulnerabilities by checking for common patterns of insecure code, such as buffer overflows, SQL injections, cross-site scripting, and weak encryption. References = What Is Static Analysis? Static Code Analysis Tools - Perforce Software, How Static Code Analysis Works | Perforce, Static Code Analysis: Techniques, Top 5 Benefits & 3 Challenges, What is MTBF? Mean Time Between Failures Explained | Perforce, Static analysis tools - Software Testing MCQs - CareerRide, ISTQB_Chapter3 | Quizizz, [Static Code Analysis for Security Vulnerabilities | Perforce].

NEW QUESTION 57

Which of the following statements about reviews are TRUE?

- A. In walkthroughs the review meeting is typically led by the autho
- B. II Inspection is characterized by an open-ended review meetingIII Preparation before the review meeting is part of informal reviews IV Management rarely participates in technical review meetings
- C. II, III
- D. I, II
- E. I, IV
- F. III, IV

Answer: C

Explanation:

The following statements about reviews are true:

? I) In walkthroughs the review meeting is typically led by the author. A walkthrough is a type of review that has a predefined objective and agenda but no formal process or roles. A walkthrough is typically led by the author of the work product under review, who guides the participants through a scenario and solicits feedback.

? IV) Management rarely participates in technical review meetings. A technical review is a type of review that has a predefined objective and agenda but no formal process or roles. A technical review is typically performed by peers with technical expertise in order to evaluate technical aspects of a work product. Management

rarely participates in technical review meetings, as they may not have sufficient technical knowledge or skills to contribute effectively. The following statements about reviews are false:

- ? II) Inspection is characterized by an open-ended review meeting. An inspection is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. An inspection is characterized by a structured review meeting with a fixed duration and agenda.
- ? III) Preparation before the review meeting is part of informal reviews. Preparation before the review meeting is part of formal reviews, such as inspections or technical reviews. Preparation involves checking

NEW QUESTION 58

Which of the following BEST matches the attributes with a level of testing?

- A. Stubs and drivers are often usedII The test environment should correspond to the production environment III Finding defects is not the main focusIV Testing can be based on use casesV Testing is normally performed by testersVI Testing for functional and non-functional characteristics
- B. Component - VI Integration - IV System - I Acceptance - I I
- C. Component - IV Integration - I System - VI Acceptance - V
- D. Component-I Integration - V System - II Acceptance - IV
- E. Component - V Integration - II System - IV Acceptance - VI

Answer: D

Explanation:

The relationship between impact analysis and regression testing in maintenance testing is that impact analysis is used to evaluate the amount of regression testing to be performed. Maintenance testing is a type of testing that is performed on an existing software product after it has been delivered or deployed, in order to ensure that it still meets its requirements and functions correctly after a change or a modification. Maintenance testing can be triggered by various reasons, such as corrective maintenance (fixing defects), adaptive maintenance (adapting to new environments), perfective maintenance (improving performance), preventive maintenance (avoiding future problems), etc. Impact analysis is a technique that is used to assess the extent and nature of changes introduced by maintenance activities on the software product or project. Impact analysis helps to identify which parts of the software product are affected by the changes, which parts need to be modified or updated accordingly, which parts need to be retested or verified for correctness or compatibility, etc. Regression testing is a type of testing that verifies that previously tested software still performs correctly after a change or a modification. Regression testing helps to detect any side effects or unintended consequences of maintenance activities on the software product's functionality or quality. Regression testing can be performed at various levels and scopes depending on the impact analysis results. Therefore, in maintenance testing, impact analysis is used to evaluate the amount of regression testing to be performed. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 20.

NEW QUESTION 63

A software company decides to invest in reviews of various types. The thought process they have is that each artifact needs to be reviewed using only one of the review methods depending on the criticality of the artifact.

- A. The thought process is incorrect
- B. The whole company should adopt same standard for review of all artifacts.
- C. The thought process is correct
- D. The whole company should decide on the review method based on their CMM level.
- E. The thought process is incorrect
- F. Same artifact can be reviewed using different review methods
- G. The thought process is correct
- H. It wastes time to review same artifact using different review methods

Answer: C

Explanation:

The thought process of the software company is incorrect, because it assumes that each artifact can be reviewed using only one review method, and that the review method depends solely on the criticality of the artifact. This is a simplistic and rigid approach that does not consider the benefits and limitations of different review methods, the context and purpose of the review, and the feedback and improvement opportunities that can be gained from multiple reviews. According to the CTFL 4.0 Syllabus, the selection of review methods should be based on several factors, such as the type and level of detail of the artifact, the availability and competence of the reviewers, the time and budget constraints, the expected defects and risks, and the desired outcomes and quality criteria. Moreover, the same artifact can be reviewed using different review methods at different stages of the development lifecycle, to ensure that the artifact meets the changing requirements, standards, and expectations of the stakeholders. For example, a requirement specification can be reviewed using an informal review method, such as a walkthrough, to get an initial feedback from the users and developers, and then using a formal review method, such as an inspection, to verify the completeness, correctness, and consistency of the specification. Therefore, the software company should adopt a more flexible and context-sensitive approach to selecting and applying review methods for different artifacts, rather than following a fixed and arbitrary rule. References = CTFL 4.0 Syllabus, Section 3.2.1, page 31-32; Section 3.2.2, page 33-34; Section 3.2.3, page 35-36.

NEW QUESTION 64

Which of the following BEST distinguishes the terms "validation" and "verification"?

- A. Verification is confirmation through the provision of objective evidence that the specified requirements have been met while validation is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- B. Verification is confirmation through the provision of subjective evidence that the specified requirements have been met while validation is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met
- C. Validation is confirmation through the provision of objective evidence that the specified requirements have been met while verification is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- D. Validation is confirmation through the provision of subjective evidence that the specified requirements have been met while verification is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met

Answer: A

Explanation:

In the context of software testing, the ISTQB Certified Tester Foundation Level (CTFL) v4.0 differentiates between "validation" and "verification" based on their respective focuses in the software development lifecycle. Verification is the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. In simpler terms, verification is about checking the product against the specified requirements to ensure it was built correctly. Validation, on the other hand, involves evaluating a system or component during or at the end of the development process to determine whether it meets specified requirements for its intended use. This means validation is about ensuring the product fulfills its intended use and

meets the needs of the user.

References:

? ISTQB CTFL Syllabus v4.0: ISTQB Official Website

? ISTQB Foundation Level Resources v4.0: ASTQB Resources

NEW QUESTION 65

"Statement Testing" is part of;

- A. Specification Based testing
- B. Decision Testing
- C. Experience based testing
- D. Structured based testing

Answer: D

Explanation:

Statement Testing is a type of white-box testing technique where the test cases are designed based on the implementation of the software, specifically aiming to execute every statement in the code at least once. This falls under the category of structure-based testing (also known as white-box testing), where the internal structure of the system is used to design test cases. Therefore, option D is correct.

NEW QUESTION 66

Which of the following BEST defines risk level?

- A. Risk level is determined by calculating the absolute value of the sum of all potential issues that may occur on the project
- B. Risk level is calculated by adding the probabilities of all planned risks to a project
- C. Risk level is calculated by dividing the sum of all known risks by the sum of all unknown risks
- D. Risk level is determined by the likelihood of an event happening and the impact or harm from that event

Answer: D

Explanation:

Risk level is determined by the combination of two factors: the likelihood of an event occurring and the impact or harm that could result from that event. This approach allows risks to be prioritized based on their potential effect on the project or system. The likelihood represents the probability of the risk event occurring, while the impact represents the severity of the consequences if the event does happen. This concept is fundamental in risk-based testing and helps guide decision-making during the testing process. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.1.

NEW QUESTION 67

Which of the following is the main benefit of a configuration management of testware?

- A. All testware is backed up with restore option, including incident reports and change request
- B. The testware can be traced to information in requirements tools and to the bug tracking system.
- C. All testware items are identified, version controlled, tracked for changes with relation to each other
- D. There is an easy way to assess the level of test coverage provided by the existing tests

Answer: C

Explanation:

Configuration management of testware is a critical aspect of maintaining the integrity and traceability of test assets throughout the testing lifecycle. The main benefit of configuration management is to ensure that all testware items, such as test cases, test scripts, test data, and test results, are systematically identified, version controlled, and tracked for changes in relation to each other.

Option C accurately describes this benefit. By applying configuration management principles to testware, teams can manage changes to test assets efficiently, ensuring that the testware remains consistent, up-to-date, and aligned with the version of the software under test. This control mechanism facilitates the reproducibility of tests, enhances the reliability of testing activities, and supports traceability from requirements through to defects.

Options A, B, and D describe other aspects of test management and testing processes but do not capture the core benefit of configuration management of testware, which is centered on the systematic control and tracking of testware items.

NEW QUESTION 68

Which of the following BEST describes checklist-based testing?

- A. Checklist-based testing includes formal tests from detailed lists of test conditions, allowing much repeatability
- B. Checklist-based testing may involve a list of tests based on what is important to the user as well as an understanding of why and how software fails
- C. Checklist-based testing, while popular, provides little consistency and few guidelines concerning test case development
- D. Checklist-based testing is restricted to non-functional testing, including usability, performance, and security test

Answer: B

Explanation:

Checklist-based testing involves using checklists that contain items, such as potential test conditions, that should be tested. These checklists are often based on insights into what is important to the user, potential areas where software might fail, and specific aspects that need to be tested. It provides a structured yet flexible approach to testing, ensuring key areas are covered while allowing testers to use their experience and understanding of the system. Checklist-based testing is not limited to non-functional testing but can be applied to various types of testing, including functional testing. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4.5.

NEW QUESTION 70

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 72

4 equivalence classes are given for integer values:

$0 < x < 100$

$100 \leq x \leq 200$

$200 < x < 500$

$x \geq 500$

Which of the following options represent correct set of data for valid equivalence class partitions?

- A. 50; 100; 200; 1000
- B. 0; 1.99; 100; 200; 201; 499; 500;
- C. 0.50; 100; 150; 200; 350; 500;
- D. 50; 100; 250; 1000

Answer: C

Explanation:

The correct set of data for valid equivalence class partitions should include one value from each equivalence class, and no value from outside the range. Option C satisfies this condition, as it has one value from each of the four equivalence classes (50, 100, 250, 500). Option A has two values from the same equivalence class (100 and 200), option B has values outside the range (0 and 0.99), and option D has two values from the same equivalence class (1000 and 500). Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 35.

NEW QUESTION 74

Which sequence of states in the answer choices is correct in accordance with the following figure depicting the life-cycle of a defect?



- A. S0->S1->S2->S3->S5->S1
- B. S0->S1->S2->S3->S5->S1->S2->S3
- C. S0->S1->S2->S3->S4
- D. S0->S1->S2->S3->S5->S3->S4

Answer: D

Explanation:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0, the life cycle of a defect typically follows a sequence from its discovery to its closure. In the provided figure, it starts with S0 (New), moves to S1 (Assigned), then to S2 (Resolved), followed by S3 (Verified). If the defect is not fixed, it can be Re-opened (S5) and goes back for verification (S3). Once verified, it is Closed (S4). References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Section 1.4.3, Page 17.

NEW QUESTION 75

Which of the following is an example of black-box dynamic testing?

- A. Functional Testing
- B. Code inspection
- C. Checking memory leaks for a program by executing it
- D. Coverage analysis

Answer: A

Explanation:

Functional testing is an example of black-box dynamic testing. Black-box testing (also known as specification-based testing) is a type of testing that does not consider the internal structure or implementation of the system under test, but rather its external behavior or functionality. Dynamic testing is a type of testing that involves executing the system under test with various inputs and observing its outputs. Functional testing is a type of black-box dynamic testing that verifies that the system under test performs its intended functions according to its requirements or specifications. Functional testing can be performed at various levels and scopes depending on the objectives and criteria of testing. The other options are not examples of black-box dynamic testing. Code inspection is an example of white-box static testing. White-box testing (also known as structure-based testing) is a type of testing that considers the internal structure or implementation of the system under test. Static testing is a type of testing that does not involve executing the system under test, but rather analyzing it for defects, errors, or violations of

standards. Code inspection is a type of white-box static testing that involves examining the source code of the system under test for quality, readability, maintainability, etc. Checking memory leaks for a program by executing it is an example of white-box dynamic testing. Memory leaks are defects that occur when a program fails to release memory that it has allocated but no longer needs. Checking memory leaks for a program by executing it requires knowledge and access to the internal structure or implementation of the program, such as memory allocation and deallocation mechanisms, pointers, references, etc. Coverage analysis is an example of white-box static testing. Coverage analysis is a technique that measures how much of the code or structure of the system under test has been exercised by a test suite. Coverage analysis requires knowledge and access to the internal structure or implementation of the system under test, such as statements, branches, paths, conditions, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 7.

NEW QUESTION 76

The following open incident report provided: Date: 01.01.01

Description: When pressing the stop button the application status remain in "Attention" instead of "Ready".

Severity: High

Life Cycle: Integration

Which of the following details are missing in the giving incident report?

- A. Identification or configuration of the applicationI
- B. The name of the developerII
- C. Recommendation of the developerIV The actions and/or conditions that came before the pressing of the button
- D. IV
- E. IV
- F. II
- G. II, III

Answer: B

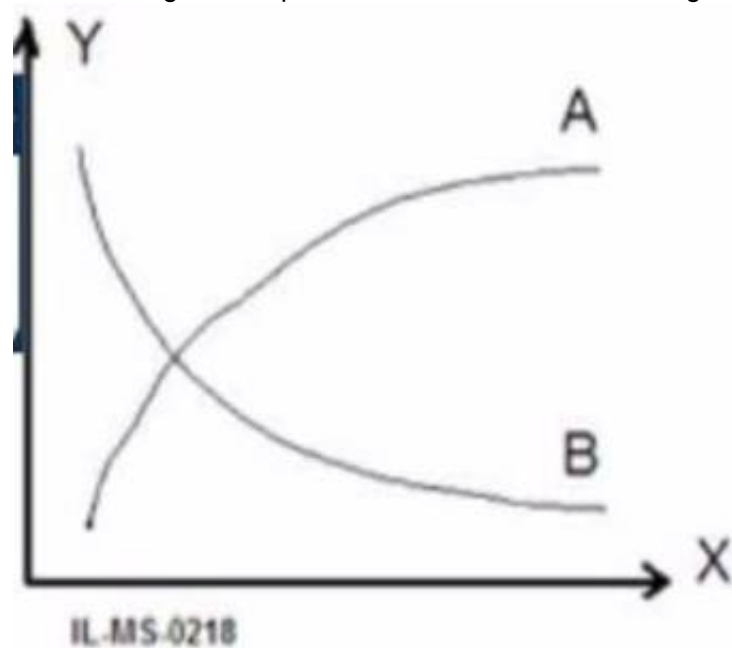
Explanation:

In an incident report, essential details provide context and facilitate the investigation and resolution of the incident. The missing elements in the given incident report are:

I. Identification or configuration of the application: This detail is crucial as it specifies which version or configuration of the application is affected, helping in reproducing the issue. IV. The actions and/or conditions that came before pressing the button: Understanding the sequence of actions leading to the issue is vital for replicating and diagnosing the problem. The name of the developer (II) and the recommendation of the developer (III) are not typically included in an incident report as they do not contribute to identifying or resolving the incident. The focus is on the incident's details, reproduction steps, and the system's state rather than on personnel or proposed solutions at this stage. Therefore, option B, which includes both I and IV, is the correct answer.

NEW QUESTION 79

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



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 83

Which of the following sentences describe a product risk?

- A. The application might not be able to provide the expected responsiveness under a load of up-to 300 concurrent users
- B. Failure in acquiring an adequate and test automation tool
- C. A wrong configuration of the test environment that causes incidents related to the environment and not to the software under test
- D. The development team lacks knowledge of the technology on which the product is based

Answer: A

Explanation:

This question relates to identifying product risks, which are potential problems associated with the product itself, such as software functionality, reliability, usability, and performance. Option A describes a scenario where the application might not meet performance requirements under specific conditions (up to 300 concurrent users), which directly impacts the product's ability to perform its intended function. This is a classic example of a product risk, as it concerns the product's quality and its ability to meet user needs. Options B, C, and D, on the other hand, relate to project risks, which are concerns related to the management and execution of the project, such as tool acquisition, environment configuration, and team expertise, rather than the quality of the product itself.

NEW QUESTION 86

Which of the following statements about test estimation approaches is CORRECT?

- A. The Wideband Delphi estimation technique is an example of the risk-based approach
- B. The Wideband Delphi estimation technique is an example of the expert-based approach
- C. Burndown charts used in Agile development is an example of the risk-based approach
- D. Burndown charts used in Agile development is an example of the expert-based approach

Answer: B

Explanation:

There are two main approaches to test estimation:

? Expert-based approach:

? Metrics-based approach:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, the expert-based approach relies on experts' experience and knowledge, which aligns with the Wideband Delphi technique's source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 87

The following 4 equivalence classes are given:

$x \leq -100$
 $-100 < x < 100$
 $100 \leq x < 1000$
 $x \geq 1000$

Which of the following alternatives includes correct test values for x. based on equivalence partitioning?

- A. -100; 100:1000; 1001
- B. -500; 0; 100; 1000
- C. -99; 99:101; 1001
- D. -1000; -100; 100; 1000

Answer: D

Explanation:

? The question is about selecting the correct test values for x based on equivalence partitioning. Equivalence partitioning is a software test design technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In this case, the given equivalence classes are:

Option D provides a value from each of these partitions:

? For $(x \leq -100)$, it gives -1000.

? For $(-100 < x < 100)$, it gives -100 and 100.

? For $(100 \leq x < 1000)$, it gives 500.

? For $(x \geq 1000)$, it gives 1500.

So, option D covers all four given equivalence classes with appropriate values. References: ISTQB Certified Tester Foundation Level (CTFL) v4.0 documents available at ISTQB and ASTQB.

? 1: ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 38

? 2: ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 39

? : ISTQB Foundation Level Syllabus 2018, Version 4.0, p. 40

NEW QUESTION 91

The testers in company A were part of the development team. Due to an organizational change they moved to be part of the support team.

What are the advantages and the disadvantages of this change?

- A. Advantage: More independence in deciding what and how to test, Disadvantage: Isolation from the development team knowledge
- B. Advantage: being closer to customer perspective, Disadvantage less independence in perspectives
- C. Advantage: pulled to support tasks and having less time for testing, Disadvantage less chances to move a tester to development
- D. Advantage: increased chances to move a tester to development; Disadvantage: pulled to support tasks and having less time for testing

Answer: B

Explanation:

Being part of the support team means that the testers are closer to the customer perspective, which is an advantage for testing, as they can better understand the user needs and expectations, and identify more realistic scenarios and risks. However, being part of the support team also means that they have less independence in deciding what and how to test, as they may be influenced by the customer's preferences or requests, which could compromise the objectivity and effectiveness of testing. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 6.

NEW QUESTION 93

Which statement about use case testing is true?

- A. The test cases are designed to find defects in the data flow.
- B. The test cases are designed to be used by real users, not by professional testers
- C. The test cases are always designed by customers or end users.
- D. The test cases are designed to find defects in the process flow.

Answer: D

Explanation:

Use case testing is a technique that helps identify test cases that exercise the whole system on a transaction by transaction basis from start to finish. Use cases are descriptions of how users interact with the system to achieve a specific goal. Use case testing is not focused on data flow, but rather on process flow. Use case testing can be performed by professional testers, customers or end users, depending on the context. Use case testing does not require the test cases to be designed by customers or end users, but rather by anyone who has access to the use case specifications. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 36.

NEW QUESTION 96

Which of the following is a key characteristic of informal reviews?

- A. Kick-off meeting
- B. Low cost
- C. Individual preparation
- D. Metrics analysis

Answer: B

Explanation:

A key characteristic of informal reviews is low cost. Informal reviews are a type of review that does not follow a formal process or have any formal documentation. Informal reviews are usually performed by individuals or small groups of peers or colleagues who have some knowledge or interest in the product under review. Informal reviews can be done at any time and for any purpose, such as checking for errors, clarifying doubts, sharing ideas, etc. Informal reviews have low cost, as

they do not require much time, effort, or resources to conduct. The other options are not key characteristics of informal reviews. Kick-off meeting is a characteristic of formal reviews, such as inspections or walkthroughs. Kick-off meeting is a meeting that is held before the review process starts, where the roles and responsibilities of the participants are defined, the objectives and scope of the review are agreed, and the logistics and schedule of the review are planned. Individual preparation is a characteristic of formal reviews, such as inspections or walkthroughs. Individual preparation is an activity that is performed by the reviewers before the review meeting, where they examine the product under review and identify any issues or questions that need to be discussed or resolved during the review meeting. Metrics analysis is a characteristic of formal reviews, such as inspections or walkthroughs. Metrics analysis is an activity that is performed after the review process is completed, where the data and results of the review are collected and analyzed to measure the effectiveness and efficiency of the review, as well as to identify any improvement actions or lessons learned for future reviews. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 9.

NEW QUESTION 98

A system computes prices for bus tickets. The price depends on

- the passenger type (baby, child, adult, senior citizen, student, military)
- the travelling type (as single or in a group)
- the distance (zone 1. 2. 3)
- the kind of transport (ordinary, express)

Which of the following test techniques is the most appropriate one for testing the price computation?

- A. Statement coverage
- B. State transition testing
- C. Equivalence partitioning
- D. Use case testing

Answer: C

Explanation:

Equivalence partitioning is a technique that divides the input data and output results of a software component into partitions of equivalent data. Each partition should contain data that is treated in the same way by the component. Equivalence partitioning can be used to reduce the number of test cases by selecting one representative value from each partition. Equivalence partitioning is suitable for testing the price computation, as it can identify different partitions based on the passenger type, the travelling type, the distance and the kind of transport. Equivalence partitioning is not statement coverage, which is a technique that measures how many executable statements in a source code are executed by a test suite. Statement coverage is not appropriate for testing the price computation, as it does not consider the input data or output results. Equivalence partitioning is not state transition testing, which is a technique that models how a system transitions from one state to another depending on events or conditions. State transition testing is not relevant for testing the price computation, as it does not involve any states or transitions. Equivalence partitioning is not use case testing, which is a technique that tests how users interact with a system to achieve a specific goal. Use case testing is not applicable for testing the price computation, as it does not focus on a single function or component. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 37-38.

NEW QUESTION 100

Which of the following BEST matches the descriptions with the different categories of test techniques?

- * 1. Test cases are based on the test basis which may include the requirements, use cases and user stories
 - * 2. Test cases are based on the test basis which may include the software architecture or code
 - * 3. Test cases can show deviations from the requirements
 - * 4. These test techniques are applicable to both functional and non-functional testing
 - * 5. Tests are based on knowledge of developers, users and other stakeholders
- Black - Black-box test techniques
White * White-box test techniques
Experience - Experience-based test techniques

- A. Black -1,3,4 White - 2 Experience - 5
- B. Black - 2, 3 White -1 Experience - 4, 5
- C. Black -1,2 White - 3, 4 Experience - 5
- D. Black - 2, 3 White -1,5 Experience - 4

Answer: A

Explanation:

Matching the descriptions with the test techniques:

? Black-box test techniques use the external description of the software, including requirements, use cases, and user stories.

? White-box test techniques use the internal structure of the software system, including software architecture and code.

? Black-box test techniques can reveal deviations from the requirements as they validate the external behavior of the software.

? Both black-box and white-box test techniques are applicable to functional and non- functional testing.

? Experience-based test techniques rely on the knowledge and intuition of developers, users, and other stakeholders (ISTQB Main Web).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

NEW QUESTION 102

You are testing the download process of a mobile phone application.

For which to the following capabilities to the system you need to design a nonfunctional test?

- A. It was easy to locate, download and install the application
- B. The application was correctly downloaded
- C. The application created an installation log file in a given folder
- D. The application installed only after the user's approval

Answer: A

Explanation:

This question asks for a non-functional aspect of testing the download process of a mobile application. Option A, "It was easy to locate, download and install the application," refers to usability, which is a non-functional quality attribute. Non-functional testing involves testing the system's attributes, such as usability, performance, reliability, etc., rather than specific behaviors or functions. Options B, "The application was correctly downloaded," C, "The application created an installation log file in a given folder," and D, "The application installed only after the user's approval," describe functional aspects, focusing on what the software does rather than how it performs or is experienced by the user.

NEW QUESTION 105

Which of the following is MOST likely to be an example of a PROJECT risk?

- A. A computation is not always performed correctly in some situations
- B. A system architecture may not support some non-functional requirements
- C. Team members' skills may not be sufficient for the assigned work
- D. Specific modules do not adequately meet their intended functions according to the user

Answer: C

Explanation:

A project risk relates to potential issues that could affect the overall success of the project. Among the options provided, the risk that "Team members' skills may not be sufficient for the assigned work" is clearly a project risk because it pertains to the potential failure of the project due to inadequate skillsets among the team. This risk affects the entire project's ability to meet its objectives. References:
? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.2.

NEW QUESTION 109

A system has a self-diagnostics module that starts executing after the system is reset. The diagnostics are running 12 different tests on the systems memory hardware. The following is one of the requirements set for the diagnostics module:

'The time taking the diagnostics tests to execute shall be less than 2 seconds' Which of the following is a failure related to the specified requirement?

- A. The diagnostic tests fail to start after a system reset
- B. The diagnostic tests take too much time to execute
- C. The diagnostic tests that measure the speed of the memory, fail
- D. The diagnostic tests fail due to incorrect implementation of the test code

Answer: B

Explanation:

A failure is an event in which a component or system does not perform a required function within specified limits¹. A requirement is a condition or capability needed by a user to solve a problem or achieve an objective². In this case, the requirement is that the diagnostics tests should execute in less than 2 seconds. Therefore, any event that violates this requirement is a failure. The only option that clearly violates this requirement is B. The diagnostic tests take too much time to execute. If the diagnostic tests take more than 2 seconds to complete, then they do not meet the specified limit and thus fail. The other options are not necessarily failures related to the specified requirement. Option A. The diagnostic tests fail to start after a system reset is a failure, but not related to the time limit. It is related to the functionality of the self-diagnostics module. Option C. The diagnostic tests that measure the speed of the memory, fail is also a failure, but not related to the time limit. It is related to the accuracy of the memory tests. Option D. The diagnostic tests fail due to incorrect implementation of the test code is also a failure, but not related to the time limit. It is related to the quality of the test code. References = ISTQB® Certified Tester Foundation Level Syllabus v4.0, Requirements Engineering Fundamentals.

NEW QUESTION 111

Which of the following is a valid collection of equivalence classes for the following problem: An integer field shall contain values from and including 1 to and including 15.

- A. Less than 0.1 through 14. 15 and more
- B. Less than 1.1 through 14. more than 15
- C. negative number
- D. 1 through 15. above 15
- E. Less than 1.1 through 15. more than 15

Answer: D

Explanation:

Equivalence partitioning is a black-box test design technique where inputs to the software or system are divided into groups that are expected to exhibit similar behavior. For an integer field that should accept values from 1 to 15, the valid equivalence class is 1 through 15. The invalid equivalence classes are numbers less than 1 and numbers more than 15. Therefore, option D, "Less than 1, 1 through 15, more than 15," correctly identifies the valid equivalence class along with the two invalid classes, covering all possible input scenarios for the field. Options A, B, and C either do not accurately capture the valid range or incorrectly specify the range boundaries.

NEW QUESTION 116

Which statement best describes the key difference between a mindset for test activities and a mindset for development activities?

- A. A tester possesses professional pessimism while a developer is concerned with validating the product
- B. A tester is concerned with finding defects while a developer is interested in designing solutions
- C. A tester is interested in building solutions while a developer is concerned with verifying the product
- D. A tester is concerned with verifying the product while a developer possesses professional pessimism

Answer: B

Explanation:

The key difference between the mindsets for test activities and development activities lies in the objectives: a tester is primarily concerned with finding defects to ensure product quality, while a developer focuses on designing and building solutions. This distinct focus helps ensure thorough quality checks and balances within the software development life cycle (ISTQB not-for-profit association). References:
? ISTQB® Certified Tester Foundation Level Syllabus v4.0: https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf

NEW QUESTION 119

Which of the following provides the BEST description of statement coverage?

- A. A white-box test technique which covers the decision results which determine the next statement to be executed

- B. A black-box test technique which uses a state table to derive test cases
- C. A white-box test technique which focuses on the percentage of executable statements that has been executed by a test suite
- D. An experience-based test technique in which test cases are based on the tester's knowledge of past failures

Answer: C

Explanation:

Statement coverage is a white-box test technique which focuses on executing all possible statements in the code at least once during testing. This helps in determining the percentage of executable statements that have been executed by the test suite, aiming to ensure that all parts of the program have been tested at least once (ISTQB Main Web).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

NEW QUESTION 121

Which of the following is NOT an experience-based technique?

- A. Boundary value analysis.
- B. Error guessing
- C. Exploratory testing
- D. Fault attack

Answer: A

Explanation:

Boundary value analysis is not an experience-based technique, but rather a specification-based technique (also known as black-box technique). Experience-based techniques are techniques that rely on the tester's knowledge and intuition to derive and select test cases based on their experience with similar systems, technologies, domains, risks, etc. Some examples of experience-based techniques are error guessing, exploratory testing, fault attack, checklist-based testing, etc. Specification-based techniques are techniques that rely on the tester's analysis and interpretation of the requirements or specifications of the system under test to derive and select test cases based on some criteria or rules. Some examples of specification-based techniques are equivalence partitioning, boundary value analysis, decision table testing, state transition testing, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 31.

NEW QUESTION 124

Which of the following CORRECTLY matches the roles and responsibilities in a formal review?

- A. Facilitator - Fixes defects in the work product under review
- B. Scribe - Collates potential defects found during the individual review activity
- C. Review Leader - Creates the work product under review
- D. Author - Identify potential defects in the work product under review

Answer: B

Explanation:

In formal reviews, the scribe's role is to collate potential defects and other findings during the review process. This position is crucial as it ensures all observations and defects are recorded accurately, facilitating efficient analysis and resolution of issues identified during the review. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 3.2.4 "Roles and Responsibilities in a Formal Review".

NEW QUESTION 127

Which of the following is a possible reason for introducing a defect in software code?

- A. Rushing to meet a tight deadline to turn code over for testing
- B. Improper unit testing
- C. Improper system testing
- D. Focus on static testing over dynamic testing

Answer: A

Explanation:

The ISTQB CTFL syllabus identifies several causes for defects in software. One prominent reason, as highlighted in the curriculum, is the pressure and rush to meet tight deadlines, which can lead to insufficiently reviewed or tested code being moved into further stages of testing or production. This scenario describes rushing to meet a deadline as a potential cause for defects because it may compromise the thoroughness of code development and testing. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.1 "Why is Testing Necessary?".

NEW QUESTION 130

Which of the following statements about re-testing and regression testing are TRUE? I Re-testing should be performed after a defect is fixed.

II Regression testing should always be performed after a defect is fixed.

III. Re-testing and regression testing may be performed at any test level.

IV Regression testing may include functional, non-functional and structural testing.

- A. Re-testing should be included in the debugging activity.
- B. I, II
- C. Iv
- D. II, v
- E. I, III
- F. I
- G. IV, V

Answer: A

Explanation:

The following statements about re-testing and regression testing are true:

? I) Re-testing should be performed after a defect is fixed. Re-testing is a type of testing that verifies that a defect has been successfully resolved by executing a test case that previously failed due to that defect. Re-testing should be performed after a defect is fixed and delivered to ensure that it does not cause any new failures or side effects.

? III) Re-testing and regression testing may be performed at any test level. Re-testing and regression testing are not limited to a specific test level, but can be applied at any level depending on the context and objectives. For example, re-testing and regression testing can be performed at unit level, integration level, system level or acceptance level.

? IV) Regression testing may include functional, non-functional and structural testing. Regression testing is a type of testing that verifies that previously tested software still performs correctly after changes. Regression testing may include various types of testing depending on the scope and purpose of the changes. For example, regression testing may include functional testing to check if the software meets its requirements, non-functional testing to check if the software meets its quality attributes, or structural testing to check if the software meets its design or code standards. The following statement about re-testing and regression testing is false:

? II) Regression testing should always be performed after a defect is fixed.

Regression testing is not always necessary after a defect is fixed, as some defects may have a low impact or low likelihood of affecting other parts of the software.

Regression testing should be performed after a defect is fixed only if there is a risk of introducing new defects or causing existing defects due to the changes made to fix the defect. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 19; Chapter 4, page 45.

NEW QUESTION 132

A Static analysis tool analyzes a given program's CONTROL FLOW among other things. Which of the following options represents the most likely outcome of the control flow analysis:

- A. Identification of unreachable code
- B. Report on adherence to the coding standards
- C. Number of comment lines
- D. Number of source code lines

Answer: A

Explanation:

A static analysis tool is a tool that analyzes a given program's source code or executable code without executing it. A static analysis tool can perform various types of analysis on a program's code, such as syntax checking, data flow analysis, control flow analysis, complexity measurement, coding standards compliance checking, etc. Control flow analysis is a type of analysis that examines how a program's statements are executed in different paths or branches. One of the most likely outcomes of control flow analysis is identification of unreachable code, which is code that can never be executed due to logical errors or design flaws.

Unreachable code can reduce readability and maintainability of the code, as well as increase complexity and size. The other options are not outcomes of control flow analysis, but rather outcomes of other types of analysis. Report on adherence to coding standards is an outcome of coding standards compliance checking.

Number of comment lines and number of source code lines are outcomes of complexity measurement. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 8.

NEW QUESTION 133

Which of the following statements about decision tables are TRUE?

- A. Generally, decision tables are generated for low risk test items.I
- B. Test cases derived from decision tables can be used for component tests.II
- C. Several test cases can be selected for each column of the decision table.I
- D. The conditions in the decision table represent negative tests generally.
- E. III
- F. I, IV
- G. I
- H. IV
- I. I
- J. III

Answer: D

Explanation:

A decision table is a technique that shows combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects). A decision table consists of four quadrants: conditions (inputs), actions (outputs), condition entries (values) and action entries (results). The following statements about decision tables are true:

? II. Test cases derived from decision tables can be used for component tests.

Decision tables can be used to test components that have multiple inputs and outputs that depend on logical combinations of conditions. Decision tables can help cover all possible combinations or scenarios in a systematic way.

? III. Several test cases can be selected for each column of the decision table. A column of a decision table represents a unique combination of condition entries and action entries. Several test cases can be selected for each column by varying other input values or expected results that are not part of the decision table. The following statements about decision tables are false:

? I. Generally, decision tables are generated for low risk test items. Decision tables are not related to risk level, but rather to complexity level. Decision tables are generated for test items that have complex logic or multiple conditions and actions that need to be tested.

? IV. The conditions in the decision table represent negative tests generally. The conditions in the decision table represent both positive and negative tests, depending on whether they are valid or invalid inputs for the test item. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 42-43.

NEW QUESTION 137

You are testing a room upgrade system for a hotel. The system accepts three differed types of room (increasing order of luxury): Platinum. Silver and Gold Luxury. ONLY a Preferred Guest Card holder s eligible for an upgrade.

Below you can find the decision table defining the upgrade eligibility:

Conditions

Preferred Guest Card holder	YES	YES	NO	NO
Room Type	Silver	Platinum	Silver	Platinum

48	Offer upgrade to Gold Luxury	YES	NO	NO	NO
	Offer upgrade to Silver	N/A	YES	N/A	NO

What is the expected result for each of the following test cases?

Customer A: Preference Guest Card holder, holding a Silver room Customer B: Non Preferred Guest Card holder, holding a Platinum room

- A. Customer A; doesn't offer any upgrade; Customer B: offers upgrade to Gold luxury room
- B. Customer A: doesn't offer any upgrade; Customer B: doesn't offer any upgrade.
- C. Customer A: offers upgrade to Gold Luxury room; Customer B: doesn't offer any upgrade
- D. Customer A: offers upgrade to Silver room; Customer B: offers upgrade to Silver room.

Answer: C

Explanation:

According to the decision table in the image, a Preferred Guest Card holder with a Silver room is eligible for an upgrade to Gold Luxury (YES), while a non-Preferred Guest Card holder, regardless of room type, is not eligible for any upgrade (NO). Therefore, Customer A (a Preferred Guest Card holder with a Silver room) would be offered an upgrade to Gold Luxury, and Customer B (a non-Preferred Guest Card holder with a Platinum room) would not be offered any upgrade. References = The answer is derived directly from the decision table provided in the image; specific ISTQB Certified Tester Foundation Level (CTFL) v4.0 documents are not referenced.

NEW QUESTION 141

Which of the following statements describes regression testing?

- A. Retesting of a fixed defectI
- B. Testing of an already tested programII
- C. Testing of new functionality in a programI
- D. Regression testing applies only to functional testingV Tests that do not have to be repeatable, because They are only used once
- E. II, IV, V
- F. I, III, IV
- G. II
- H. 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 change1 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 change2 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 145

Which of the following is NOT a product risk?

- A. Poor software usability
- B. Failure-prone software is delivered
- C. Problems in defining the right requirements
- D. Software does not perform the intended functions

Answer: C

Explanation:

Problems in defining the right requirements is not a product risk, but rather a project risk. A product risk is a risk that affects the quality or performance of the software product itself, such as poor usability, failure-prone functionality, security vulnerabilities, compatibility issues, etc. A project risk is a risk that affects the management or delivery of the software project itself, such as unrealistic schedule, insufficient resources, unclear scope, changing requirements, etc. The other options are examples of product risks, as they relate to the software product's characteristics or features. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 12.

NEW QUESTION 149

Which of the following statements is LEAST likely to be describing component testing?

- A. It identifies defects in modules and classes.
- B. Simulators and stubs may be required.
- C. It mainly tests interfaces and interaction between components.
- D. It may be applied using a test-first approach.

Answer: C

Explanation:

Component testing (also known as unit testing or module testing) is a level of testing that focuses on verifying the functionality and quality of individual software components (such as modules, classes, functions, methods, etc.). Component testing mainly tests interfaces and interaction between components, as well as internal logic and data structures of the components. Component testing may be applied using a test-first approach (such as test-driven development or behavior-driven development), where tests are written before the code is implemented. Component testing does not identify defects in modules and classes, as this is a result of component testing, not an objective. Simulators and stubs may be required for component testing, as they can simulate or replace missing or incomplete components or external systems that are needed for testing. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 19.

NEW QUESTION 152

Can "cost" be regarded as Exit criteria?

- A. Ye
- B. Spending too much money on test ng will result in an unprofitable product, and having cost as an exit criterion helps avoid this
- C. N
- D. The financial value of product quality cannot be estimated, so it is incorrect to use cost as an exit criterion
- E. Ye
- F. Going by cost as an exit criterion constrains the testing project which will hello 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 153

Which of the following statements about independent testing is WRONG?

- A. Independent testing is necessary because developers don't know any testing.
- B. Independent testing is best suited for the system test level.
- C. A certain degree of independence makes the tester more effective at finding defects.
- D. Independent test teams may find other types of detects than developers who are familiar with the system's structure.

Answer: A

Explanation:

Independent testing is testing performed by a person or group that is independent of the development team. Independent testing can have various degrees of independence, ranging from testers who are part of the same organization as developers to testers who are external contractors or consultants. Independent testing can have various benefits, such as reducing bias, increasing objectivity, improving quality, or providing different perspectives. Independent testing is not necessary because developers don't know any testing, as this is a wrong and disrespectful statement. Developers can perform various types of testing, such as unit testing, component testing, or integration testing. However, independent testing can complement developer testing by providing additional levels of verification and validation, such as system testing, acceptance testing, or non- functional testing. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 16-17.

NEW QUESTION 155

An Incident Management tool implements the following defect states; Open, Assigned, Solved,

Closed Consider the following defect report: Id T000561

Test Object "Warehouse Management" application Tester name; John Bishop

Date: 10th. April 2010 Test Case MRT558I

Status OPEN Severity Serious Priority

Problem- After inputting the Total Quantity item = 450 in the SV034 screen, the system shows an unexpected Error message=47

Correction: Developer name: Closing date:

Which of the following is a valid criticism of this report?

- A. The Priority, the Correction description and the Developer name are missing
- B. The version of the application is missing
- C. There is no link to the applicable requirement (traceability)
- D. The description is not highlighting the source of the problem

Answer: B

Explanation:

A valid criticism of this report is that the version of the application is missing. The version of the application is an important piece of information that should be included in a defect report, as it helps to identify which release or build of the software product contains the defect. The version of the application can also help to reproduce and debug the defect, as different versions may have different behaviors or features. The other options are not valid criticisms of this report. The priority, the correction description and the developer name are not missing, but rather not applicable for this report. The priority is a measure of how urgently a defect needs to be fixed, which can be assigned by the project manager or the defect tracking system, not by the tester who reports the defect. The correction description and the developer name are information that are added after the defect has been resolved, not when it has been reported. There is no link to the applicable

requirement (traceability) is not a valid criticism of this report, because traceability is not a mandatory attribute of a defect report, but rather an optional one. Traceability is a relationship between two or more entities (such as requirements, test cases, defects, etc.) that shows how they are related or dependent on each other. Traceability can help to verify that the requirements are met by the test cases and defects, but it is not essential for reporting a defect. The description is not highlighting the source of the problem is not a valid criticism of this report, because highlighting the source of the problem is not a responsibility of the tester who reports the defect, but rather of the developer who fixes the defect. The description should provide enough information to describe what happened when the defect occurred, such as input values, expected results, actual results, error messages, screenshots, etc., but it does not need to explain why or how it happened. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 140.

NEW QUESTION 159

Your manager asked you when testing will be complete. In order to answer this question, you'll most likely use:

- A. Test progress reports
- B. Your colleagues advice
- C. A conversion spreadsheet
- D. A Test Oracle

Answer: A

Explanation:

When a manager asks when testing will be complete, the most appropriate and informative resource to provide an answer is test progress reports (Option A). Test progress reports contain detailed information on the status of testing activities, including what has been accomplished, what remains to be done, the results of the tests conducted, and any issues or risks that might impact the completion of testing. These reports allow for an informed assessment of the testing progress and estimation of when testing might be completed. Options B, C, and D do not provide the structured, detailed, and specific information required to accurately answer the manager's question about the completion of testing.

NEW QUESTION 164

The ISTOB glossary defines Quality Assurance as: "Pail or quality management focused on providing confidence that quality requirements will be fulfilled. Which of the following Is not one of the Quality Assurance activity?"

- A. Requirements elicitation
- B. Defect analysis
- C. Functional Testing
- D. Performance Testing

Answer: C

Explanation:

Quality Assurance (QA) activities are focused on providing confidence that quality requirements will be fulfilled through planned and systematic processes. These activities are preventive in nature, aimed at ensuring quality is built into the product from the beginning.

? Requirements elicitation (A) is part of the requirements engineering process and is concerned with gathering the needs and conditions to meet for a new or altered product.

? Defect analysis (B) can be part of QA activities as it involves analyzing defects to prevent them in future development cycles.

? Functional Testing (C) and Performance Testing (D) are types of dynamic testing, which are actually Quality Control activities rather than Quality Assurance. They are concerned with the identification of defects in the product, not with the processes to prevent defects.

Since the question asks for an activity that is NOT part of Quality Assurance, options A and B are incorrect because they can be part of QA activities. Between C and D, while both are dynamic testing activities, Functional Testing (C) is more directly related to verifying the functionality against specified requirements, which is more aligned with Quality Control. Therefore, C is the best answer.

NEW QUESTION 166

You are responsible for applying the correct technique for a review of the requirements document for a project to develop a new software application. You identify the reviewers and the required roles, including the meeting leader, who is the requirements document author, and a separate role for a scribe. Additionally, you decide to take a relatively informal approach to the requirements review. The goal of the review is to find defects in the requirements document, such as omissions, Inconsistencies, and duplications. Another goal of the review is to improve the software application's usability and accessibility by considering the various stakeholders' viewpoints.

Which of the following statements BEST describes this scenario?

- A. This scenario is using a pair review type and a perspective-based review technique
- B. This scenario is using a walkthrough review type and a checklist-based review technique
- C. This scenario is using a walkthrough review type and a perspective-based review technique
- D. This scenario is using a pair review type and a checklist-based review technique

Answer: C

Explanation:

This scenario is using a walkthrough review type and a perspective-based review technique. In a walkthrough, the author of the document leads the meeting and it typically includes a meeting leader and a scribe, as described. This type of review is informal, focuses on discussion, and often involves scenario-based reading of the document to understand different user perspectives (ISTQB Main Web).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

NEW QUESTION 167

Which of the following is correct with regards to debugging?

- A. Debugging identifies the cause of a failure
- B. Debugging is often performed by test engineers
- C. Debugging is considered part of the testing activities
- D. Debugging is intended to find as many defects as possible in the code

Answer: A

Explanation:

Debugging is the process of finding, analyzing and removing the causes of failures in software. Debugging is not considered part of testing, but rather a development activity that can involve testing. Debugging is not intended to find as many defects as possible, but rather to fix the specific failure that was observed. Debugging is usually performed by developers, not by test engineers. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 1, page 6.

NEW QUESTION 172

A software calculates the annual car tax using three inputs:

- E; the emission level of the vehicle
- P: the power of the vehicle
- T the type of the vehicle

The input value for P can be integer positive values between 15 and 350.

Which of the following answers contains a correct list of a boundary values for the P input?

- A. 14,351
- B. 14,15,350,351
- C. 15,350
- D. 5.175.500

Answer: B

Explanation:

A correct list of boundary values for the P input should include the minimum and maximum values of the valid range (15 and 350), as well as the values just below and above the boundaries (14 and 351). Boundary value analysis is a test design technique that involves testing the values at or near the boundaries of an input domain or output range, as these values are more likely to cause errors than values in the middle. Option B satisfies this condition, as it has all four boundary values (14, 15, 350, 351). Option A has only two boundary values (14 and 351), option C has only two boundary values (15 and 350), and option D has no boundary values at all. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 34.

NEW QUESTION 174

A QA manager of a start-up company needs to implement within a week a low cost incident management tool. Which of the following is the best option?

- A. Document incidents on a large board in the lab
- B. Purchase and deploy an incident management tool
- C. Manage the incidents through E-mails and phone calls
- D. Manage the incidents in a spreadsheet posted on the intranet

Answer: D

Explanation:

An incident is any event that occurs during testing that requires investigation. An incident management tool is a software tool that supports recording and tracking incidents throughout their life cycle. A QA manager of a start-up company needs to implement within a week a low cost incident management tool. The best option for this case is to manage the incidents in a spreadsheet posted on the intranet. This option has several advantages over other options:

? It is low cost, as it does not require purchasing any additional software or hardware.

? It is easy to implement within a week, as it does not require installing or configuring any complex software or hardware.

? It is accessible and transparent, as it can be viewed and updated by anyone who has access to the intranet.

? It is structured and organized, as it can store and display various information about incidents, such as identifier, summary, description, severity, priority, status, resolution, etc. The other options are not suitable for this case, as they have several disadvantages over the chosen option:

? Documenting incidents on a large board in the lab is not a good option, as it is not accessible or transparent to anyone who is not physically present in the lab. It is also not structured or organized, as it may not store or display all the necessary information about incidents.

? Purchasing and deploying an incident management tool is not a good option, as it is not low cost or easy to implement within a week. It may require spending a significant amount of money and time on acquiring, installing and configuring the software or hardware.

? Managing the incidents through emails and phone calls is not a good option, as it is not structured or organized. It may lead to confusion, inconsistency or loss of information about incidents. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 32-33.

NEW QUESTION 177

In what way do Configuration Management effects testing?

- A. Without proper configuration management, test planning cannot proceed.
- B. Proper configuration management ensures that testers can uniquely identify the tested item
- C. Configuration management is important for developers, not for testers
- D. There is very little influence of configuration management practices on the test project.

Answer: B

Explanation:

Configuration management is a process that establishes and maintains consistency among work products throughout their life cycle. Configuration management affects testing in various ways, such as:

? Proper configuration management ensures that testers can uniquely identify the tested item, which can help traceability, reproducibility and accountability.

? Proper configuration management ensures that testers have access to consistent versions of software components and testware, which can help reliability, compatibility and efficiency.

? Proper configuration management ensures that testers can track changes and defects in software components and testware, which can help verification, validation and reporting.

? Proper configuration management ensures that testers can control the configuration of the test environment, which can help stability, security and performance.

Configuration management is not a prerequisite for test planning, as test planning can proceed without configuration management, although it may be less effective or accurate. Configuration management is not important for developers only, but for testers as well, as it affects the quality and consistency of the testing process and products. Configuration management has a significant influence on the test project, as it affects various aspects of testing, such as traceability, reproducibility, reliability, compatibility, efficiency, verification, validation, reporting, stability, security and performance. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 6, page 60-61.

NEW QUESTION 181

Which of the following is an example of the absence-of-errors fallacy?

- A. Repeating the same test cases will continue to find new defects, even after dozens of executions
- B. A small number of modules contains the most defects
- C. Since testing found very few defects, the system certainly will be successful
- D. Other than trivial cases, it is not feasible to test all combinations of inputs and preconditions

Answer: C

Explanation:

The absence-of-errors fallacy, as explained in the ISTQB syllabus, is the erroneous belief that having fewer defects found in testing equates to a system being more successful upon release. This misconception can lead stakeholders to undervalue thorough testing. Answer C illustrates this fallacy perfectly: assuming that because few defects were found, the system will be successful, neglects the many other factors that contribute to system success, including user satisfaction and fit-for-purpose. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.1.5 "Absence-of-errors fallacy".

NEW QUESTION 186

Which of the following is the most correct statement about state testing techniques?

- A. Static techniques can be used before all code is ready for execution
- B. Static techniques find more defects than dynamic techniques.
- C. Static techniques can be used by inexperienced users.
- D. Static techniques are always cheaper than dynamic techniques.

Answer: A

Explanation:

State testing techniques are a type of dynamic testing techniques that are based on the behavior of the system under test for different input conditions and events. Dynamic testing techniques require the system to be executed with test cases, whereas static testing techniques do not. Static testing techniques can be applied before the code is ready for execution, such as reviews, inspections, walkthroughs, and static analysis. Static testing techniques can help find defects early in the development process, improve the quality of the code, and reduce the cost and effort of dynamic testing. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 4, Section 4.2.1, Page 281; ISTQB Glossary of Testing Terms v4.0, Page 292

NEW QUESTION 187

A bank offers a savings account with various interest rates based on the current balance in the account. The balance ranges and respective interest rates are:

Up to \$100.00 = 2%

\$100.01 to \$500.00 = 4%

\$500.01 to \$1,000.00 = 5% Above \$1,000.00 = 7%

Using two-point boundary value analysis, which of the following sets of test inputs provides the relatively highest level of boundary coverage?

- A. \$5.00, \$100.00, \$499.99, \$1,000.00, \$1,000.01
- B. \$100.00, \$100.01, \$100.02, \$500.00, \$999.99
- C. \$100.00, \$500.00, \$1,000.00, \$1,000.01
- D. \$5.00, \$100.00, \$500.00, \$1,000.01

Answer: B

Explanation:

Boundary Value Analysis (BVA) is a software testing technique in which tests are designed to include values at the boundaries. The concept is to focus on the boundaries since errors tend to occur at the edges of input ranges rather than in the middle.

Given the problem statement:

? Up to \$100.00 = 2%

? \$100.01 to \$500.00 = 4%

? \$500.01 to \$1,000.00 = 5%

? Above \$1,000.00 = 7%

Two-point boundary value analysis means testing the two boundaries of each range. For each range:

? The boundaries for "Up to \$100.00" would be \$100.00 and \$100.01.

? The boundaries for "\$100.01 to \$500.00" would be \$100.00 and \$500.00.

? The boundaries for "\$500.01 to \$1,000.00" would be \$500.00 and \$1,000.00.

? The boundaries for "Above \$1,000.00" would be \$1,000.00 and \$1,000.01. Now, let's examine the options:

? A. \$5.00, \$100.00, \$499.99, \$1,000.00, \$1,000.01

? B. \$100.00, \$100.01, \$100.02, \$500.00, \$999.99

? C. \$100.00, \$500.00, \$1,000.00, \$1,000.01

? D. \$5.00, \$100.00, \$500.00, \$1,000.01

Given the options, B provides the highest boundary coverage (ISTQB not-for-profit association) (Udemy).

References:

? Certified Tester Foundation Level v4.0

? 10 Sample Exams ISTQB Foundation Level (CTFL) v4.0

NEW QUESTION 190

Which of the following statements is CORRECT?

- A. Test cases are made up of input values, expected results and actual results developed to cover test objectives
- B. Test cases describe items or events to test that are derived from the test basis during the test analysis activity
- C. Test cases are sequences of actions for test execution specified during the test implementation activity
- D. Test cases are derived during the test design activity to cover test objectives or test conditions

Answer: C

Explanation:

A test case is a set of input values, execution preconditions, expected results and execution postconditions, developed for a particular objective or test condition. A test case is a sequence of actions for test execution that can be followed by a tester or a test automation tool. A test case is specified during the test implementation activity, which is the activity that prepares the testware needed for test execution. A test case does not include actual results, as these are obtained during test execution and compared with the expected results. A test case does not describe items or events to test, as these are derived from the test basis during the test analysis activity. A test case is not derived during the test design activity, as this is the activity that specifies the test conditions or objectives that need to be tested. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 23-24; Chapter 4, page 34.

NEW QUESTION 192

Which of the following lists factors That contribute to PROJECT risks?

- A. skill and staff shortages; problems in defining the right requirements, contractual issues.
- B. skill and staff shortages; software does not perform its intended functions; problems in defining the right requirements.
- C. problems in defining the right requirements; contractual issues; poor software quality characteristics.
- D. poor software quality characteristics; software does not perform its intended functions.

Answer: A

Explanation:

Project risks are the uncertainties or threats that may affect the project objectives, such as scope, schedule, cost, and quality. According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, some of the factors that contribute to project risks are:

? Skill and staff shortages: This factor refers to the lack of adequate or qualified human resources to perform the project tasks. This may result in delays, errors, rework, or low productivity.

? Problems in defining the right requirements: This factor refers to the difficulties or ambiguities in eliciting, analyzing, specifying, validating, or managing the requirements of the project. This may result in misalignment, inconsistencies, gaps, or changes in the requirements, affecting the project scope and quality.

? Contractual issues: This factor refers to the challenges or disputes that may arise from the contractual agreements between the project parties, such as clients, suppliers, vendors, or subcontractors. This may result in legal, financial, or ethical risks, affecting the project delivery and satisfaction.

The other options are not correct because they list factors that contribute to PRODUCT risks, not project risks. Product risks are the uncertainties or threats that may affect the quality or functionality of the software product or system. Some of the factors that contribute to product risks are:

? Poor software quality characteristics: This factor refers to the lack of adherence or compliance to the quality attributes or criteria of the software product or system, such as reliability, usability, security, performance, or maintainability. This may result in defects, failures, or dissatisfaction of the users or stakeholders.

? Software does not perform its intended functions: This factor refers to the deviation or discrepancy between the expected and actual behavior or output of the software product or system. This may result in errors, faults, or malfunctions of the software product or system.

References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, Chapter 1: Fundamentals of Testing, Section 1.5: Risks and Testing, Pages 14-16.

NEW QUESTION 193

Which of the following statements is an example of testing contributing to higher quality?

- A. A test leader writes a test summary report
- B. A project manager asks to a test leader to estimate the test effort
- C. A tester installs a test ten in the lest environment
- D. A tester finds a bug which is resolved prior to release

Answer: D

Explanation:

? The question is about identifying an example of testing contributing to higher quality. Quality is the degree to which a component, system or process meets specified requirements and/or user/customer needs and expectations¹. Testing is the process consisting of all lifecycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects².

? Therefore, testing contributes to higher quality by verifying and validating that the software products and related work products meet the specified requirements, are fit for purpose and have no defects, or at least have a reduced number of defects. Testing also provides information about the quality of the software products and related work products to the stakeholders, who can make informed decisions based on the test results³.

? Out of the four given statements, only option D is an example of testing contributing to higher quality, as it shows that testing has detected a defect (a flaw in a component or system that can cause the component or system to fail to perform its required function⁴) and that the defect has been resolved (fixed and confirmed) prior to release (delivery of the software product to the customer or end user). This means that testing has prevented a potential failure (an event in which a component or system does not perform a required function within specified limits) from occurring in the operational environment, and thus has improved the quality of the software product.

? Option A is not an example of testing contributing to higher quality, as it is a reporting activity that summarizes the test results and evaluates the test objectives, but does not directly affect the quality of the software product or related work products. A test summary report is a document that records and communicates the outcomes of testing activities, including test completion criteria, test results, incident reports, test summary and evaluation, and lessons learned.

? Option B is not an example of testing contributing to higher quality, as it is a planning activity that estimates the resources and time needed for testing activities, but does not directly affect the quality of the software product or related work products. A test effort estimate is an approximation of the amount of work and/or the duration of time required to perform testing activities.

? Option C is not an example of testing contributing to higher quality, as it is a preparation activity that sets up the test environment (an environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test), but does not directly affect the quality of the software product or related work products. A test environment installation is a process of installing and configuring the test environment according to the test environment specification. References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 10

? 2: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 11

? 3: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 12

? 4: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 77

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 78

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 79

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 80

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 81

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

? : 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
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 86
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 87
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 88
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 89
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 90
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 91
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 92
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 93
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 94
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 95
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 96
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 97
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 98
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 99
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 100
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 101
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 102
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 103
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 104
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 105
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 106
? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 107

NEW QUESTION 195

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

ISTQB-CTFL Practice Exam Features:

- * ISTQB-CTFL Questions and Answers Updated Frequently
- * ISTQB-CTFL Practice Questions Verified by Expert Senior Certified Staff
- * ISTQB-CTFL Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * ISTQB-CTFL Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The ISTQB-CTFL Practice Test Here](#)