



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 is a correct reason to apply test automation?

- A. When a new test automation tool is launched
- B. When there are a lot of repetitive testing tasks
- C. When it is easy to automate
- D. When it is cheap to buy test automation tools

Answer: B

Explanation:

A correct reason to apply test automation is when there are a lot of repetitive testing tasks. Test automation is the use of software tools or scripts to perform or support testing activities, such as test case execution, test result comparison, test data generation, etc. Test automation can be beneficial when there are a lot of repetitive testing tasks that need to be performed frequently or consistently, such as regression testing, performance testing, load testing, etc. Test automation can help to save time and effort, increase reliability and accuracy, and improve coverage and efficiency of testing. The other options are not correct reasons to apply test automation. When a new test automation tool is launched is not a reason to apply test automation, but rather a factor for choosing a test automation tool. When it is easy to automate is not a reason to apply test automation, but rather a factor for evaluating the feasibility of test automation. When it is cheap to buy test automation tools is not a reason to apply test automation, but rather a factor for estimating the cost and benefit of test automation. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 10.

NEW QUESTION 3

Which of the following statements about testing in the context of an agile (iterative- incremental) development model is correct?

- A. Unit test and acceptance test are the most important tests to make sure that the system works as expected.
- B. Each iteration of testing has to be completely finished before a new iteration of development starts.
- C. Regression testing is necessary whenever a new increment is added to the existing system.
- D. Only certain types of non-functional and explorative testing are performed.

Answer: C

Explanation:

In the context of agile (iterative-incremental) development models, testing is integrated into the development process and occurs continuously throughout the lifecycle of the project. Agile testing emphasizes adaptability and the need for feedback at various stages of development. Option C is correct because regression testing is indeed necessary whenever a new increment is added to the existing system. Agile development often involves frequent changes and additions to the codebase, which can potentially introduce new defects into previously tested code. Regression testing ensures that new changes have not adversely affected existing functionality. Options A, B, and D present misconceptions about agile testing:
? A is incorrect because, in agile, all types of testing (unit, integration, system, acceptance) are important and occur throughout the iteration, not just unit and acceptance tests.
? B is incorrect because agile methodologies advocate for continuous integration and testing, where development and testing activities overlap and support each other throughout an iteration.
? D is incorrect because agile methodologies encourage a wide range of testing types, including both functional and non-functional, as well as exploratory testing, to ensure a comprehensive quality assessment.

NEW QUESTION 4

Which of the following would be LEAST appropriate as part of an incident report covering the observation of a failure during testing?

- A. SQL injection into the username entry field allowed a variety of SQL commands to be executed by the application without the appropriate authority.
- B. The user interface was complicated and confusing and I found it quite difficult to follow the test script.
- C. The updates made as part of the 'add new member' function did not reflect the expected change as the name was written into the address field.

D. The expected result for the 'list friends' response time was less than 10 seconds, whereas the average response time obtained was 13 seconds.

Answer: B

Explanation:

An incident report during testing should focus on factual observations of failures or defects in the system, including their impacts and how they deviate from expected results. Options A, C, and D describe specific issues that are directly related to the system's behavior or performance and are suitable for inclusion in an incident report. Option B, which describes the user interface as "complicated and confusing" and relates to the tester's personal difficulty in following the test script, is more subjective and relates to the tester's experience rather than an objective observation of a system failure. Therefore, option B is the least appropriate for an incident report.

NEW QUESTION 5

Manager responsibilities in formal review includes ad except one of the following:

- A. Planning the review
- B. Determines if the review objectives have been met
- C. Decide on the execution of reviews
- D. Allocate time for review

Answer: B

Explanation:

A formal review is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. A formal review can have various roles involved, such as manager, moderator, author, reviewer and scribe. The manager responsibilities in formal review include all except one of the following:

- ? Planning the review (correct responsibility)
 - ? Determines if the review objectives have been met (incorrect responsibility)
 - ? Decide on the execution of reviews (correct responsibility)
 - ? Allocate time for review (correct responsibility)
- The responsibility of determining if the review objectives have been met belongs to the moderator role, not to the manager role. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 3, page 28-29.

NEW QUESTION 6

The four test levels used in ISTQB syllabus are:

- * 1. Component (unit) testing
- * 2. Integration testing
- * 3. System testing
- * 4. Acceptance testing

An organization wants to do away with integration testing but otherwise follow V-model. Which of the following statements is correct?

- A. It is allowed as organizations can decide on men test levels to do depending on the context of the system under test
- B. It is allowed because integration testing is not an important test level arc! can be dispensed with.
- C. It is not allowed because integration testing is a very important test level and ignoring i: means definite poor product quality
- D. It is not allowed as organizations can't change the test levels as these are chosen on the basis of the SDLC (software development life cycle) model

Answer: D

Explanation:

The V-model is a software development life cycle model that defines four test levels that correspond to four development phases: component (unit) testing with component design, integration testing with architectural design, system testing with system requirements, and acceptance testing with user requirements. The V-model emphasizes the importance of verifying and validating each phase of development with a corresponding level of testing, and ensuring that the test objectives, test basis, and test artifacts are aligned and consistent across the test levels. Therefore, an organization that wants to follow the V-model cannot do away with integration testing, as it would break the symmetry and completeness of the V-model, and compromise the quality and reliability of the software or system under test. Integration testing is a test level that aims to test the interactions and interfaces between components or subsystems, and to detect any defects or inconsistencies that may arise from the integration of different parts of the software or system. Integration testing is essential for ensuring the functionality, performance, and compatibility of the software or system as a whole, and for identifying and resolving any integration issues early in the development process. Skipping integration testing would increase the risk of finding serious defects later in the test process, or worse, in the production environment, which would be more costly and difficult to fix, and could damage the reputation and credibility of the organization. Therefore, the correct answer is D.

The other options are incorrect because:

- ? A. It is not allowed as organizations can decide on the test levels to do depending on the context of the system under test. While it is true that the choice and scope of test levels may vary depending on the context of the system under test, such as the size, complexity, criticality, and risk level of the system, the organization cannot simply ignore or skip a test level that is defined and required by the chosen software development life cycle model. The organization must follow the principles and guidelines of the software development life cycle model, and ensure that the test levels are consistent and coherent with the development phases. If the organization wants to have more flexibility and adaptability in choosing the test levels, it should consider using a different software development life cycle model, such as an agile or iterative model, that allows for more dynamic and incremental testing approaches.
- ? B. It is not allowed because integration testing is not an important test level and can be dispensed with. This statement is false and misleading, as integration testing is a very important test level that cannot be dispensed with. Integration testing is vital for testing the interactions and interfaces between components or subsystems, and for ensuring the functionality, performance, and compatibility of the software or system as a whole. Integration testing can reveal defects or inconsistencies that may not be detected by component (unit) testing alone, such as interface errors, data flow errors, integration logic errors, or performance degradation. Integration testing can also help to verify and validate the architectural design and the integration strategy of the software or system, and to ensure that the software or system meets the specified and expected quality attributes, such as reliability, usability, security, and maintainability. Integration testing can also provide feedback and confidence to the developers and stakeholders about the progress and quality of the software or system development. Therefore, integration testing is a crucial and indispensable test level that should not be skipped or omitted.
- ? C. It is not allowed because integration testing is a very important test level and ignoring it means definite poor product quality. This statement is partially true, as integration testing is a very important test level that should not be ignored, and skipping it could result in poor product quality. However, this statement is too strong and absolute, as it implies that integration testing is the only factor that determines the product quality, and that ignoring it would guarantee a poor product quality. This is not necessarily the case, as there may be other factors that affect the product quality, such as the quality of the requirements, design, code, and other test levels, the effectiveness and efficiency of the test techniques and tools, the competence and experience of the developers and testers, the availability and adequacy of the resources and environment, the management and communication of the project, and the expectations and satisfaction of the customers and users. Therefore, while integration testing is a very important test level that should not be skipped, it is not the only test level that matters, and skipping it does not necessarily mean definite poor product quality, but rather a higher risk and likelihood of poor product quality.

References = ISTQB Certified Tester Foundation Level Syllabus, Version 4.0, 2018, Section 2.3, pages 16-18; ISTQB Glossary of Testing Terms, Version 4.0, 2018, pages 38-39; ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 104, page 36.

NEW QUESTION 7

Which of the following is a function of a dynamic analysis tool?

- A. Provide support for traceability of tests, test results and incidents to source documents
- B. Monitor the allocation, use and de-allocation of memory during run-time of a program
- C. Execute programs step-by-step in order to reproduce failures and find corresponding defects
- D. Provide support for release of baselines consisting of configuration items

Answer: B

Explanation:

A dynamic analysis tool is a tool that performs analysis of a software product based on its behavior during execution. A dynamic analysis tool can monitor various aspects of a program's run-time performance, such as memory usage, CPU load, response time, or resource leaks. A dynamic analysis tool can monitor the allocation, use and de-allocation of memory during run-time of a program, which can help detect defects such as memory leaks, buffer overflows, or memory corruption. A dynamic analysis tool cannot provide support for traceability of tests, test results and incidents to source documents, as this is a function of a test management tool. A dynamic analysis tool cannot execute programs step-by-step in order to reproduce failures and find corresponding defects, as this is a function of a debugging tool. A dynamic analysis tool cannot provide support for release of baselines consisting of configuration items, as this is a function of a configuration management tool. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 6, page 56-57.

NEW QUESTION 8

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 9

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 10

A test engineer finds a defect while testing. After the developer has fixed the defect, the test engineer decides to re-run a complete section of the tests. Which of the following is correct?

- A. The test engineer should not re-run the tests, as they have already been run, and results recorded.
- B. The test engineer should not re-run the tests, they should be part of the developer tests.
- C. The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix.
- D. The test engineer should re-run the tests, because the defect shows that the test cases need to be updated.

Answer: C

Explanation:

The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix. This is also known as regression testing, which is a type of testing that verifies that previously tested software still performs correctly after a change. Regression testing helps to detect any side effects or unintended consequences of a fix or a modification. The other options are incorrect reasons for re-running the tests. The test engineer should not re-run the tests, as they have already been run, and results recorded, because this ignores the possibility of new defects caused by the fix. The test engineer should not re-run the tests, they should be part of the developer tests, because this assumes that developer tests are sufficient and reliable, which may not be true. The test engineer should not re-run the tests, because the defect shows that the test cases need to be updated, because this does not address the impact of the fix on other test cases or functionalities. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 41.

NEW QUESTION 10

A test score indicator for students produces a performance score based on a combination of the number of consecutive hours studied (below 4 hours, 4 to 8 hours, 9 to 12 hours or above 12 hours) and the average intensity of focus on the material during the study time (low, medium or high).

Given the following test cases: hours intensity score

T1 3 low55

T2 14 high 95

T3 9 low75

What is the minimum number of additional test cases that are needed to ensure full coverage of all valid INPUT equivalence partitions?

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

Answer: C

Explanation:

Considering the various valid input equivalence partitions of hours studied and intensity, three additional test cases are needed to fully cover all valid partitions. This would typically include testing combinations that vary both the number of hours and the intensity levels not covered by the initial test cases (ISTQB Main Web).References:

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

NEW QUESTION 15

A class grade application for instructors assigns letter grades based on students' numerical grades.

The letter grades for different numerical grades should be:

Above 89, up to 100 - A

Above 79, up to 89 • B

Above 69, up to 79 • C

Above 59, up to 69 - D

Below 60- F

Which of the following sets of test inputs would achieve the relatively highest equivalence partition coverage?

- A. 0, 58.59,70, 80
- B. 74, 79.84,85, 89
- C. 79, 89.90,99, 100
- D. 69, 79. 80, 89, 90

Answer: D

Explanation:

The set of test inputs that achieve the relatively highest equivalence partition coverage for grading students is option D: 69, 79, 80, 89, 90. This set effectively tests the boundaries between each grade category, ensuring that the grading system accurately transitions from one grade to another at the correct thresholds (ISTQB Main Web)

.References:

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

NEW QUESTION 16

Given the following review process main activities and specific review activities:

* a. Planning

* b. Initiate review

* c. Issue communication and analysis d.Fixing and reporting

* 1. Creating defect reports

* 2. Estimating effort and timeframe

* 3. Recording updated status of defects

* 4. Selecting the people to participate

* 5. Distributing the work product and other material

* 6. Evaluating the review findings

Which of the following BEST matches the review process main activities with the appropriate specific review activities?

- A. 2-a, 4-a, 5-b, 6-c, 1-d, 3-d
- B. 2-a, 5-a, 1-b, 4-b, 3-c, 6-d
- C. 1-a, 4-b, 5-b, 6-c, 2-d, 3-d
- D. 2-a, 4-b, 5-c, 1-
- E. 3-d, 6-d

Answer: A

Explanation:

Matching the main review process activities with the specific review activities, we see that planning includes estimating effort and timeframe (2) and selecting people to participate (4). Initiating a review involves distributing work products and other material (5). Issue communication and analysis includes evaluating the review findings (6). Fixing and reporting would entail creating defect reports (1) and recording the updated status of defects (3).References:ISTQB Certified Tester

Foundation Level Syllabus v4.0, Section 3.2 "Review Process".

NEW QUESTION 17

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 20

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 25

In the newest version of payroll system number of changes were made. As a tester you got a task to perform regression and confirmation tests. Which of the following project activities are related to confirmation testing?

- A. Testing due to the application of a new version of the interface
- B. Testing that fixes resolved the defects in the search function
- C. Testing if a system still works after update of an operating system
- D. Testing to ensure the adding of a new functionalities haven't broken existing functions

Answer: B

Explanation:

Confirmation testing, also known as re-testing, is performed to verify that specific defects have been successfully fixed.

Option A: "Testing due to the application of a new version of the interface" would typically involve regression testing, not confirmation testing.

Option B: "Testing that fixes resolved the defects in the search function" fits the description of confirmation testing as it focuses on ensuring that specific issues have been addressed.

Option C: "Testing if a system still works after update of an operating system" is an example of regression testing, as it checks the overall system behavior after an update.

Option D: "Testing to ensure the adding of new functionalities haven't broken existing functions" is another example of re (ISTQB

not-for-profit association) (Udemy) it checks for unintended consequences of new changes.

Therefore, the correct answer is B. References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 28

ST is a Software Testing organization which utilizes a testing knowledge base. Access to ST knowledge base can be either full or limited. Access level is determined based on ST certification and testing experience as follows:

- * 1. If ST certified, with less than 5 years testing experience - allow limited access
- * 2. If ST certified, 5-10 years of testing experience - allow full access
- * 3. If not ST certified with 5-10 years of testing experience - allow limited access.

What would be the results for:

- A - ST certified. 12 years of testing experience
- B - Not ST certified. 7 years of testing experience
- C - Not ST certified. 3 years of testing experience

- A. A - unknown B - limited access C- unknown
- B. A - full access B - limited access C - unknown
- C. A - full access B - limited access C - limited access

D. A - unknown B - full access C - unknown

Answer: B

Explanation:

The correct answer can be derived by applying the given rules to each case:

? A is ST certified and has 12 years of testing experience, which is more than 10 years. Therefore, A does not match any of the rules and the result is unknown.

? B is not ST certified and has 7 years of testing experience, which is between 5 and 10 years. Therefore, B matches rule 3 and the result is limited access.

? C is not ST certified and has 3 years of testing experience, which is less than 5 years. Therefore, C does not match any of the rules and the result is unknown.

Verified References: This question does not require any external references, as it is based on logical reasoning.

NEW QUESTION 33

A test manager defined the following test levels in her test plan; Component, System and Acceptance.

Which Software Development lifecycle is the Test Manager most likely following?

- A. V-Model
- B. Agile
- C. Waterfall
- D. Prototyping

Answer: A

Explanation:

The test manager is most likely following the V-model for software development. The V-model is a software development model that defines four testing levels that correspond to four development phases: component testing corresponds to component design, integration testing corresponds to architectural design, system testing corresponds to system requirements specification, and acceptance testing corresponds to user requirements specification. The V-model also defines the test planning and test execution activities for each testing level. Agile is a software development model that follows an iterative and incremental approach, where testing is integrated into each iteration and adapts to changing requirements and feedback. Waterfall is a software development model that follows a sequential and linear approach, where testing is performed after the development phase is completed. Prototyping is a software development model that involves creating a simplified version of the software to elicit user feedback and validate requirements before developing the final product. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 18.

NEW QUESTION 36

Which of the following statements contradicts the general principles of testing?

- A. Most defects are found in a small subset of a system's modules.
- B. If new defects are to be found we should run the same test set more often.
- C. Testing is better if it starts at the beginning of a project.
- D. How testing is done, is based on the situation in a particular project.

Answer: B

Explanation:

Statement B contradicts the general principles of testing, because running the same test set more often will not increase the chances of finding new defects, unless there are some changes in the system or environment that affect the test results. Running different test sets with different inputs, outputs or conditions would be more effective in finding new defects. Statements A, C and D are consistent with the general principles of testing. Statement A states that most defects are found in a small subset of a system's modules, which is true according to the defect clustering principle. Statement C states that testing is better if it starts at the beginning of a project, which is true according to the early testing principle. Statement D states that how testing is done, is based on the situation in a particular project, which is true according to the context-dependent testing principle. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, pages 4-6.

NEW QUESTION 37

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 40

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

Given the following state model of sales order software: SEE ATTACHMENT
 Which of the following sequences of transitions provides the highest level of transition coverage for the model (assuming you can start in any state)?

- A. IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED
- B. IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION
- C. PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED -> PLACED
- D. PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED

Answer: B

Explanation:

State transition testing is a black-box testing technique where test cases are designed to cover states and transitions of a state machine.

Given the state model with the following transitions:

- ? PLACED -> IN PRODUCTION
- ? IN PRODUCTION -> CANCELLED
- ? IN PRODUCTION -> SHIPPED
- ? SHIPPED -> INVOICED
- ? INVOICED -> CANCELLED
- ? CANCELLED -> PLACED

To cover all transitions at least once, we need to create a sequence that covers all six transitions.

Option A: IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED- Misses SHIPPED -> INVOICED and INVOICED -> CANCELLED transitions.

Option B: IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION- Covers all transitions.

Option C: PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED (ISTQB not-for-profit association) (Udemy)sses INVOICED -> CANCELLED transition.

Option D: PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED- Misses SHIPPED -> INVOICED and INVOICED -> CANCELLED transitions.

Given these, Option B covers all the transitions. References:

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

NEW QUESTION 47

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 52

During system testing phase of a word processor, a tester finds that on opening a file from a particular set of files, which are part of a critical workflow, the word processor crashes. Which of the following is the next step the tester should take prior to recording the deviation?

- A. Try to recreate the incident before reporting
- B. Try to identify the code fragment causing the problem
- C. Send an email to the developer and not report the bug
- D. Report the incident as is without any further action

Answer: A

Explanation:

An incident is any event that occurs during testing that requires investigation. An incident report is a document that records the details of an incident. The next step the tester should take prior to recording the deviation is to try to recreate the incident before reporting. This can help confirm that the incident is reproducible and not caused by a random or external factor. This can also help gather more information about the incident, such as the steps to reproduce it, the expected and actual results, the severity and priority of the incident, or any screenshots or logs that can illustrate the incident. Trying to identify the code fragment causing the problem is not the next step the tester should take prior to recording the deviation, as this is a debugging activity that is usually performed by developers after receiving the incident report. Sending an email to the developer and not reporting the bug is not the next step the tester should take prior to recording the deviation, as this is an informal and unstructured way of communicating incidents that can lead to confusion, inconsistency or loss of information. Reporting the incident as is without any further action is not the next step the tester should take prior to recording the deviation, as this can result in incomplete or inaccurate incident reports that can hamper the investigation and resolution of incidents. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 32-33.

NEW QUESTION 54

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 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 - 111
- 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 58

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 incorrec
- B. The whole company should adopt same standard for review of all artifacts.
- C. The thought process is correc
- D. The whole company should decide or the review method based on their CMM level.
- E. The thought process is incorrec
- F. Same artifact can be reviewed using different review methods
- G. The thought process is correc
- H. It wastes time to review same artifact using efferent 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 61

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

Which of the following statements about estimation of the test effort is WRONG?

- A. Once the test effort is estimated, resources can be identified and a schedule can be drawn up.
- B. Effort estimate can be inaccurate because the quality of the product under tests is not known.
- C. Effort estimate depends on the budget of the project.
- D. Experience based estimation is one of the estimation techniques.

Answer: C

Explanation:

? Effort estimate does not depend on the budget of the project, but rather on the scope, complexity, and quality of the software product and the testing activities¹. Budget is a constraint that may affect the feasibility and accuracy of the effort estimate, but it is not a factor that determines the effort estimate. Effort estimate is the amount of work required to complete the testing activities, measured in terms of person-hours, person-days, or person-months².

? The other options are correct because: References =

? 1 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 154

? 2 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 155

? 3 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 156

? 4 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 157

? 5 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 158

? 6 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 159

? 7 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 16

? [8] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 160

? [9] ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 161

NEW QUESTION 67

"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 71

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 74

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 78

Which of the following tasks is MOST LIKELY to be performed by the tester?

- A. Develop a test strategy and test policy for the organization
- B. Promote and advocate the test team within the organization
- C. Create the detailed test execution schedule
- D. Introduce suitable metrics for measuring test progress

Answer: C

Explanation:

Testers are typically involved in creating detailed test execution schedules, among other tasks such as designing tests, executing tests, and logging defects. Creating a test strategy and test policy, promoting and advocating the test team, and introducing metrics are typically responsibilities of test managers or senior roles.

In the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, the responsibilities of testers include creating test cases, setting up test (ISTQB not-for-profit association)nts, executing tests, and reporting defects, which align with creating detailed test execution schedules6†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 79

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 81

Which of the following statements BEST describes how test cases are derived from a use case?

- A. Test cases are derived based on non-functional requirements such as usability
- B. Test cases are created using white-box test techniques to execute scenarios of use cases
- C. Test cases are derived based on pair testing between a user and a tester to find defects
- D. Test cases are designed to cover various user behaviors, including basic, exceptional or alternative and error behaviors associated with human users or systems

Answer: D

Explanation:

Use cases describe a system's behavior as it responds to a request from a user. They typically consist of various scenarios, such as basic flow, alternative flow, and exceptional flow, which represent possible behaviors when a user interacts with the system. When deriving test cases from use cases, it is important to cover these different types of user behaviors. Test cases should be designed to verify how the system behaves during each of these scenarios. This ensures that the system operates correctly for normal and error conditions encountered by human users or systems interacting with the application. Thus, test cases derived from use cases aim to cover basic, exceptional, and alternative flows, ensuring comprehensive coverage. References:
 ? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.2.4.

NEW QUESTION 84

Which of the following should be included in a test status report?

- A. Estimation detailsI
- B. Total number of open and closed defects III Actual effort spentI
- C. Defect reports
- D. Number of executed, failed, blocked tests
- E. III.V
- F. II, III
- G. I
- H. IV
- I. II, III.V

Answer: D

Explanation:

The following should be included in a test status report: total number of open and closed defects, actual effort spent, and number of executed, failed, and blocked tests.

A test status report is a document that provides information on the results and status of testing activities for a given period or phase. A test status report should include information that is relevant, accurate, and timely for the intended audience and purpose. Some of the information that should be included in a test status report are: total number of open and closed defects, which can indicate the defect trend and defect density of the software product; actual effort spent, which can indicate the productivity and efficiency of the testing process; number of executed, failed, and blocked tests, which can indicate the test progress and test coverage of the software product. The following should not be included in a test status report: estimation details, defect reports, and impact analysis. Estimation details are not part of a test status report, but rather part of a test plan or a test estimation document. Estimation details provide information on the expected time, resources, and costs for testing activities, not on the actual results or status of testing activities. Defect reports are not part of a test status report, but rather separate documents that provide detailed information on individual defects found during testing. Defect reports include information such as defect description, defect severity, defect priority, defect status, defect resolution, etc. Defect reports can be referenced or summarized in a test status report, but not included in full. Impact analysis is not part of a test status report, but rather part of a risk assessment or prioritization process. Impact analysis provides information on the potential effects or consequences of a change or a defect on the software product or project. Impact analysis can be used to evaluate the amount or scope of testing to be performed, but not to report the results or status of testing activities. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 141.

NEW QUESTION 89

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 mil 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 are1:

? 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 93

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 95

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 97

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 technique⁶†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 99

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 101

Which of the following statements is not correct?

- A. Looking for defects in a system may require ignoring system details
- B. Identifying defects may be perceived as criticism against product
- C. Looking for defects in system requires professional pessimism and curiosity
- D. Testing is often seen as a destructive activity instead of constructive activity

Answer: A

Explanation:

? Looking for defects in a system does not require ignoring system details, but rather paying attention to them and understanding how they affect the system's quality, functionality, and usability. Ignoring system details could lead to missing important defects or testing irrelevant aspects of the system.

? Identifying defects may be perceived as criticism against product, especially by the developers or stakeholders who are invested in the product's success.

However, identifying defects is not meant to be a personal attack, but rather a constructive feedback that helps to improve the product and ensure its alignment with the requirements and expectations of the users and clients.

? Looking for defects in system requires professional pessimism and curiosity, as testers need to anticipate and explore the possible ways that the system could fail, malfunction, or behave unexpectedly. Professional pessimism means being skeptical and critical of the system's quality and reliability, while curiosity means being eager and interested in finding out the root causes and consequences of the defects.

? Testing is often seen as a destructive activity instead of constructive activity, as it involves finding and reporting the flaws and weaknesses of the system, rather than creating or enhancing it. However, testing is actually a constructive activity, as it contributes to the system's improvement, verification, validation, and optimization, and ultimately to the delivery of a high-quality product that meets the needs and expectations of the users and clients.

NEW QUESTION 104

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 108

Which of the following statements is the BEST example of non-functional testing?

- A. Tests which capture the time it takes to save a file
- B. Tests which calculate overtime pay for those employees entitled to such
- C. Tests related to "what" the system should do
- D. Tests based on the internal structure of a component or system

Answer: A

Explanation:

Non-functional testing refers to testing aspects that do not relate to specific behaviors or functions of the software but to attributes such as performance, usability, reliability, etc. Tests that capture the time it takes to save a file directly relate to the performance of the system, thus falling under non-functional testing. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.2.5 "Functional and Non-functional Testing".

NEW QUESTION 109

Which of the following statements about static analysis are FALSE?

- A. Static analysis can be used Instead of dynamic testing.I
- B. Static analysis can uncover defects like security vulnerabilities.II
- C. Static analysis can be used to check conformance to specifications and standard
- D. IV Static analysis typically detects failures prior to component testing.
- E. II
- F. I
- G. III
- H. II
- I. IV
- J. I, IV

Answer: D

Explanation:

Static analysis involves analyzing the software's code, design, and structure without executing the program. It can uncover various types of defects, including security vulnerabilities (II) and non-conformance to specifications and standards (III). However, static analysis cannot replace dynamic testing (I), which involves executing the software to observe its behavior under various conditions. Dynamic testing can identify failures that static analysis cannot, such as those related to runtime issues and interaction between different parts of the software. Statement IV is false because static analysis does not detect failures; it detects defects. Failures are observed when the software is executed, which is beyond the scope of static analysis.

NEW QUESTION 113

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 117

Which of the following is NOT a deciding factor in determining the extent of testing required?

- A. Level of risk of the product or features
- B. Budget to do testing
- C. A particular tester involved in testing
- D. Time available to do testing

Answer: C

Explanation:

The extent of testing required for a software product depends on various factors, such as the level of risk, the budget, and the time available. The level of risk reflects the potential impact of failures on the stakeholders and the environment. The budget determines how much resources can be allocated for testing. The time available defines the schedule and deadlines for testing activities. The particular tester involved in testing is not a deciding factor for the extent of testing required, as testing should be based on objective criteria and not on personal preferences or abilities. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 14-15.

NEW QUESTION 120

Which of the following BEST describes a test summary report for executive-level employees

- A. The report is detailed and includes a status summary of defects by priority or budget
- B. The report is detailed and includes specific information on defects and trends
- C. The report is high-level and includes a status summary of defects by priority or budget
- D. The report is high-level and includes specific information on defects and trends

Answer: C

Explanation:

For executive-level employees, a test summary report should be concise and focus on high-level information. It typically includes a summary of defects categorized by priority or budget. Executives are generally interested in the overall status and the impact on critical business objectives rather than detailed technical information. The report should provide an overview of the most important aspects of testing, such as key issues, test progress, and any risks or concerns that could affect project outcomes. References:
 ? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 5.3.1.

NEW QUESTION 124

Given the following priorities and dependencies for these test cases: SEE ATTACHMENT

Which of the following test execution schedules BEST takes into account the priorities and technical and logical dependencies?

- A. TC1 - TC3 - TC2 - TC4 - TC6 - TC5
- B. TC3 - TC4 - TC2 - TC6 - TC1 - TC5
- C. TC1 - TC3 - TC2 - TC4 - TC5 - TC6
- D. TC2 - TC4 - TC1 - TC3 - TC5 - TC6

Answer: C

Explanation:

When scheduling test cases, priorities and dependencies must be considered. The best execution order will respect both the logical dependencies and the priorities assigned to each test case.

Given the options, the correct order considering the priorities and dependencies is:

- ? TC1 (Priority 1)
- ? **TC (ISTQB not-for-profit association)ity 2, dependent on TC1)
- ? TC2 (Priority 3, dependent on TC1)
- ? TC4 (Priority 4)
- ? TC5 (Priority 5)
- ? TC6 (Priority 6, dependent on TC4)

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, understanding dependencies and scheduling tests accordingly is crucial for effective test execution.

References:

- ? Certified Tester Foundation Level v4.0
- ? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 126

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

For which of the following capabilities of 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 128

Consider the following statements about risk-based testing.

- I) Risk-based testing has the objective to reduce the level of protect risks.
- II) Tests should be prioritized to find tie critical detects as early as possible.
- III) Non-testing activities may also help to reduce risk
- IV) Risks have to be reassessed on a regular basis.
- V) The project stakeholders can give useful input to determine the risks

- A. I III IV and V are tru
- B. II is false.
- C. II, III IV and V are correc
- D. I is false.
- E. I, II and IV are tru
- F. III and V are false.
- G. II, III and V are tru
- H. 1 ants lv are false.

Answer: B

Explanation:

The following statements about risk-based testing are correct:

? II) Tests should be prioritized to find tie critical detects as early as possible. Risk- based testing involves prioritizing tests based on risk level, which reflects both the likelihood and impact of defects or failures. Tests with higher risk level should be executed earlier than tests with lower risk level, in order to find and fix critical defects as soon as possible.

? III) Non-testing activities may also help to reduce risk. Risk-based testing does not only involve testing activities, but also other activities that can help mitigate risks, such as reviews, inspections, audits, simulations or prototyping.

NEW QUESTION 132

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 135

Which of the following is an INCORRECT statement about the benefit of traceability between the test basis and test work products?

- A. Traceability may be required by IT governance rules
- B. Traceability may help evaluate the extent of test coverage
- C. Traceability may allow testing to be auditable
- D. Traceability may make it harder to understand the impact of changes

Answer: D

Explanation:

The statement "Traceability may make it harder to understand the impact of changes" is incorrect. Traceability in testing actually facilitates understanding the impact of changes by linking test cases to requirements. This linkage helps ensure that any changes in the requirements are adequately reflected and verified in the test cases, thus supporting effective management of changes and maintaining the integrity of the system or product being developed (ISTQB not-for-profit association) (ISTQB Main Web). 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 Official Website - CTFL Certification: <https://www.istqb.org/certifications/certified-tester-foundation-level/>

NEW QUESTION 136

In which of the following cases you would NOT execute maintenance testing?

- A. Retirement of the software or system
- B. Modifications to a released software or system
- C. Migration of the system data to a replacement system
- D. Update to the Maintainability requirements during the development phase

Answer: D

Explanation:

Maintenance testing is testing performed on a software product after delivery to correct defects or improve performance or other attributes. Maintenance testing can be triggered by various situations, such as modifications to a released software or system, migration of the system data to a replacement system, or retirement of the software or system. Maintenance testing is not executed when there is an update to the maintainability requirements during the development phase, as this is not a maintenance situation but rather a change request that should be handled by the development process. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 18-19.

NEW QUESTION 137

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 140

Which of the following are the phases of the ISTQB fundamental test process?

- A. Test planning and control, Test analysis and design, Test implementation and execution, Evaluating exit criteria and reporting
- B. Test closure activities
- C. Test planning, Test analysis and design
- D. Test implementation and control
- E. Checking test coverage and reporting, Test closure activities
- F. Test planning and control, Test specification and design
- G. Test implementation and execution, Evaluating test coverage and reporting, Retesting and regression testing, Test closure activities
- H. Test planning
- I. Test specification and design
- J. Test implementation and execution
- K. Evaluating exit criteria and reporting
- L. Retesting and test closure activities

Answer: A

Explanation:

The ISTQB fundamental test process consists of five main phases, as described in the ISTQB Foundation Level Syllabus, Version 4.0, 2018, Section 2.2, page 15:

? Test planning and control: This phase involves defining the test objectives, scope, strategy, resources, schedule, risks, and metrics, as well as monitoring and controlling the test activities and results throughout the test process.

? Test analysis and design: This phase involves analyzing the test basis (such as requirements, specifications, or user stories) to identify test conditions (such as features, functions, or scenarios) that need to be tested, and designing test cases and test procedures (such as inputs, expected outcomes, and execution steps) to cover the test conditions. This phase also involves evaluating the testability of the test basis and the test items (such as software or system components), and selecting and implementing test techniques (such as equivalence partitioning, boundary value analysis, or state transition testing) to achieve the test objectives and optimize the test coverage and efficiency.

? Test implementation and execution: This phase involves preparing the test environment (such as hardware, software, data, or tools) and testware (such as test cases, test procedures, test data, or test scripts) for test execution, and executing the test procedures or scripts according to the test plan and schedule. This phase also involves logging the outcome of test execution, comparing the actual results with the expected results, and reporting any discrepancies as incidents (such as defects, errors, or failures).

? Evaluating exit criteria and reporting: This phase involves checking if the planned test activities have been completed and the exit criteria (such as quality, coverage, or risk levels) have been met, and reporting the test results and outcomes to the stakeholders. This phase also involves making recommendations for the release or acceptance decision based on the test results and outcomes, and identifying any residual risks (such as known defects or untested areas) that need to be addressed or mitigated.

? Test closure activities: This phase involves finalizing and archiving the testware and test environment for future reuse, and evaluating the test process and the test project against the test objectives and the test plan. This phase also involves identifying any lessons learned and best practices, and communicating the findings and suggestions for improvement to the relevant parties.

References = ISTQB Certified Tester Foundation Level Syllabus, Version 4.0, 2018, Section 2.2, page 15; ISTQB Glossary of Testing Terms, Version 4.0, 2018, pages 37-38;

ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 88, page 32.

NEW QUESTION 143

Which of the following activities is NOT a part of the fundamental testing process?

- A. Archiving automation code
- B. Test status reporting
- C. Test process improvement
- D. Build release and maintenance

Answer: D

Explanation:

The fundamental testing process includes activities that are directly related to the planning, preparation, execution, and evaluation of tests, as well as the closure

activities of the testing phase. Option D, "Build release and maintenance," falls outside the scope of the fundamental testing process as it relates more to software development and operations rather than specific testing activities. Options A, "Archiving automation code," B, "Test status reporting," and C, "Test process improvement," are all activities that can be part of or associated with the fundamental testing process. Archiving automation code is part of test closure, test status reporting is part of test monitoring and control, and test process improvement can be an outcome of test closure activities.

NEW QUESTION 146

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 150

For the following pseudo-code determine number of tests required for 100% statement coverage

```
IF Gender = Boy
  If Age > 3 AND Age < 5 Shoe Size = 1
  ELSE IF Age >=5 AND Age < 7
  Shoe Size = 2 ENDIF
ELSE
  IF Age > 3 AND Age < 5
  Shoe Size = 0
  ELSE IF Age >=5 AND Age < 7
  Shoe Size = 1 ENDIF ENDIF
```

- A. 6
- B. 4
- C. 2
- D. 6

Answer: B

Explanation:

To achieve 100% statement coverage, we need to design test cases that ensure every statement in the given pseudo-code is executed at least once. Analyzing the pseudo-code, we notice that there are conditions based on two variables: Gender and Age. To cover all statements, we need to consider the paths that lead to each assignment of the Shoe Size variable.

- ? Gender = Boy, Age <= 3 (Shoe Size assignment is not reached, but the condition is evaluated)
- ? Gender = Boy, Age > 3 AND Age < 5 (Shoe Size = 1)
- ? Gender = Boy, Age >= 5 AND Age < 7 (Shoe Size = 2)
- ? Gender != Boy, Age <= 3 (Again, Shoe Size assignment is not reached, but the condition is evaluated)
- ? Gender != Boy, Age > 3 AND Age < 5 (Shoe Size = 0)
- ? Gender != Boy, Age >= 5 AND Age < 7 (Shoe Size = 1)

However, upon closer inspection, we see that tests 1 and 4 do not contribute to statement coverage as they do not lead to a Shoe Size assignment. Therefore, we only need 4 test cases to achieve 100% statement coverage, making option B the correct answer.

NEW QUESTION 153

Which of the following BEST describes exploratory testing?

- A. Exploratory testing is a suitable test technique which may replace both black-box and white-box test techniques
- B. Exploratory testing is a valid and useful black-box test technique since it focuses on test cases related to the architecture and design of a system
- C. Exploratory testing requires both solid specifications and much project time available for test execution
- D. Exploratory testing may be used within defined time periods, during which the tester may follow a test charter as a guide

Answer: D

Explanation:

Exploratory testing involves simultaneous test design and execution and is guided by a test charter, which outlines what needs to be tested, how it should be tested, and what to look for. This technique is typically conducted within predefined time periods, known as time-boxing, which allows testers to explore a system, understand its functionalities, and identify potential issues without detailed documentation or prior test case planning. The key aspects of exploratory testing include flexibility, adaptability, and the ability to respond to system behavior during testing. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4.2.

NEW QUESTION 154

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 155

You are testing an e-commerce system that sporadically fails to properly manage customers' shopping carts. You have stressed the urgency of this situation to the development manager and development team and they recognize the priority of resolving the underlying defect. The development team is waiting for more information, which you will include in your defect report. Given the following items of information they are included in a typical defect report:

- * 1. The expected results
- * 2. The actual results
- * 3. The urgency and priority to fix this
- * 4. The date and author of the defect report
- * 5. A description of the defect in order to reproduce, including screenshots and database dumps

Which of these items will be MOST useful to the developers to help them identify and remove the defect causing this failure?

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

Answer: A

Explanation:

When developers are trying to identify and remove a defect, they need clear information on what went wrong and what was expected. The items that will be most useful to developers in this context are the expected results (item 1), the actual results (item 2), and a description of the defect including steps to reproduce, screenshots, and database dumps (item 5). This information helps developers understand the nature of the defect and provides the necessary details to reproduce and diagnose the issue effectively. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 5.5.1.

NEW QUESTION 158

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 defects 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 159

Which of the following is NOT a common objective of testing?

- A. Finding defects in the software
- B. Preventing defects
- C. Debugging the software to find the reason for defects
- D. Providing information on the status of the system

Answer: C

Explanation:

Debugging the software to find the reason for defects is not a common objective of testing, but rather a task of development or maintenance. Debugging is a process of locating and fixing errors in the software code, while testing is a process of finding and reporting defects in the software behavior or quality. Testing does not aim to fix defects, but rather to provide information on their existence and impact. The other options are common objectives of testing. Finding defects in the software is one of the main objectives of testing, as it helps to improve the quality and reliability of the software. Preventing defects is another objective of testing, as it helps to avoid rework and reduce costs and risks. Providing information on the status of the system is another objective of testing, as it helps to support decision making and risk management. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 3.

NEW QUESTION 164

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 169

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 173

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 177

For a mandatory input field "ZIP code" the following rules are given:

- 1 - The valid ZIP code format is 5 numeric digits.
- 2 - The code has to exist in the post office's official ZIP code list

Using equivalence classes partitioning, how many test cases are required to test this field?

- A. 8
- B. 3
- C. 6
- D. 4

Answer: D

Explanation:

Equivalence classes 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 classes partitioning can be used to reduce the number of test cases by selecting one representative value from each partition. For the ZIP code field, there are four equivalence classes based on the given rules:

? Valid ZIP code format and valid ZIP code value (e.g., 12345)

? Valid ZIP code format and invalid ZIP code value (e.g., 99999)

? Invalid ZIP code format and valid ZIP code value (e.g., 1234)

? Invalid ZIP code format and invalid ZIP code value (e.g., ABCDE) Therefore, four test cases are required to test this field, one for each equivalence class.

Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 37-38.

NEW QUESTION 179

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 181

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 183

During component testing of a program if 100% decision coverage is achieved, which of the following coverage criteria is also guaranteed to be 100%?

- A. 100% State transition coverage
- B. 100% Equivalence class coverage
- C. 100% Boundary value coverage
- D. 100% Statement coverage

Answer: D

Explanation:

Statement coverage is a structural coverage metric that measures the percentage of executable statements in the source code that are executed by a test suite¹. Decision coverage is another structural coverage metric that measures the percentage of decision outcomes (such as branches or conditions) in the source code that are executed by a test suite¹. Decision coverage is a stronger metric than statement coverage, because it requires that every possible outcome of each decision is tested, while statement coverage only requires that every statement is executed at least once². Therefore, if a test suite achieves 100% decision coverage, it also implies that it achieves 100% statement coverage, because every statement in every branch or condition must have been executed. However, the converse is not true: 100% statement coverage does not guarantee 100% decision coverage, because some branches or conditions may have multiple outcomes that are not tested by the test suite². For example, consider the following pseudocode:

```
if (x > 0) then print("Positive") else print("Non-positive") end if
```

A test suite that executes this code with $x = 1$ and $x = -1$ will achieve 100% statement coverage, because both print statements are executed. However, it will not achieve 100% decision coverage, because the condition $x > 0$ has only been tested with two outcomes: true and false. The third possible outcome, $x = 0$, has not been tested by the test suite. Therefore, the test suite may miss a potential bug or error in the condition or the branch.

The other options, such as state transition coverage, equivalence class coverage, and boundary value coverage, are not guaranteed to be 100% by achieving 100% decision coverage. State transition coverage is a structural coverage metric that measures the percentage of transitions between states in a state machine that are executed by a test suite³. Equivalence class coverage is a functional coverage metric that measures the percentage of equivalence classes (or partitions) of input or output values that are tested by a test suite⁴. Boundary value coverage is another functional coverage metric that measures the percentage of boundary values (or extreme values) of input or output ranges that are tested by a test suite⁴. These metrics are independent of decision coverage, because they are based on different aspects of the system under test, such as its behavior, functionality, or specification. Therefore, achieving 100% decision coverage does not imply achieving 100% of any of these metrics, and vice versa. References = ISTQB® Certified Tester Foundation Level Syllabus v4.0, Test Coverage in Software Testing -Guru99, Structural Coverage Metrics - MATLAB & Simulink - MathWorks India, Test Design Coverage in Software Testing - GeeksforGeeks.

NEW QUESTION 185

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 187

What type of testing measures its effectiveness by tracking which lines of code were executed by the tests?

- A. Acceptance testing
- B. Structural testing
- C. Integration testing
- D. Exploratory testing

Answer: B

Explanation:

Structural testing is a type of testing that measures its effectiveness by tracking which lines of code were executed by the tests. Structural testing, also known as white-box testing or glass-box testing, is based on the internal structure, design, or implementation of the software. Structural testing aims to verify that the software meets the specified quality attributes, such as performance, security, reliability, or maintainability, by exercising the code paths, branches, statements, conditions, or data flows. Structural testing uses various coverage metrics, such as function coverage, line coverage, branch coverage, or statement coverage, to determine how much of the code has been tested and to identify any untested or unreachable parts of the code. Structural testing can be applied at any level of testing, such as unit testing, integration testing, system testing, or acceptance testing, but it is more commonly used at lower levels, where the testers have access to the source code.

The other options are not correct because they are not types of testing that measure their effectiveness by tracking which lines of code were executed by the tests. Acceptance testing is a type of testing that verifies that the software meets the acceptance criteria and the user requirements. Acceptance testing is usually performed by the end-users or customers, who may not have access to the source code or the technical details of the software. Acceptance testing is more concerned with the functionality, usability, or suitability of the software, rather than its internal structure or implementation. Integration testing is a type of testing that verifies that the software components or subsystems work together as expected. Integration testing is usually performed by the developers or testers, who may use both structural and functional testing techniques to check the interfaces, interactions, or dependencies between the components or subsystems. Integration testing is more concerned with the integration logic, data flow, or communication of the software, rather than its individual lines of code. Exploratory testing is a type of testing that involves simultaneous learning, test design, and test execution. Exploratory testing is usually performed by the testers, who use their creativity, intuition, or experience to explore the software and discover any defects, risks, or opportunities for improvement. Exploratory testing is more concerned with the behavior, quality, or value of the software, rather than its internal structure or implementation. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, Chapter 4: Test Techniques, Section 4.3: Structural Testing Techniques, Pages 51-54; Chapter 1: Fundamentals of Testing, Section 1.4: Testing Throughout the Software Development Lifecycle, Pages 11-13; Chapter 3: Static Testing, Section 3.4: Exploratory Testing, Pages 40-41.

NEW QUESTION 188

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 193

Software was found to take much more time than the stated requirement of less than one second to save a file. Upon investigation it was found that there was an unnecessary check inside a loop which was slowing down the file-save operation. The software not being able to meet the desired response time is an example of

- A. It is not a defect
 B. Defect
 C. Error
 D. Failure

Answer: D

Explanation:

A failure is an event in which a component or system does not perform a required function within specified limits. A failure is observable by the software users or other stakeholders. A failure is caused by one or more defects in the software. In this case, the software not being able to meet the desired response time is an example of a failure, as it deviates from the stated requirement and affects the user experience. It is not a defect, which is a flaw in the software that causes the failure. It is not an error, which is a human action that produces an incorrect result. It is not a non-defect, as it clearly violates a specified requirement. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 4.

NEW QUESTION 196

Given the following examples of entry and exit criteria:

- * 1. A defined level of code coverage has been achieved
- * 2. The test automation tool has been installed and properly configured
- * 3. The number of unresolved defects is within the predefined limit
- * 4. The performance test environment has been set-up and is available
- * 5. The user stories have proper acceptance criteria defined
- * 6. The testing budget has been spent and the project sponsor bears the risk of not testing any further

Which of the following BEST categorizes them as entry and exit criteria:

- A. Entry criteria - 2, 4, 5 Entry criteria -1, 3, 4
 B. Entry criteria - 2, 4 Entry criteria - 2, 4, 5, 6
 C. Exit criteria -1,3,6 Exit criteria - 2, 5, 6
 D. Exit criteria -1,3,5,6 Exit criteria -1,3

Answer: A

Explanation:

Entry and exit criteria are used to determine when to start and stop testing, respectively.

? Entry Criteria:

? Exit Criteria:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, these criteria help in effectively managing the testing process⁶†source.

References:

- ? Certified Tester Foundation Level v4.0
 ? ISTQB Foundation Level Syllabus 4.0 (2023)

NEW QUESTION 200

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 test 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 202

.....

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](#)