



iSQI

Exam Questions CTFL-AT

Certified Tester Foundation Level Agile Tester

NEW QUESTION 1

Which of the following statements would you expect to be the MOST direct advantage of the whole-team approach?

- A. Having at least once a day an automated build and test process that detects integration errors early and quickly.
- B. Avoiding requirements misunderstandings which may not have been detected until later in the development cycle when they are more expensive to fix.
- C. Capitalizing on the combined skills of business representatives, testers and developers working together to contribute to project success.
- D. Reducing the involvement of business representatives because of the increased communication and collaboration between testers and developers.

Answer: C

Explanation:

The whole-team approach is a principle of agile testing that involves everyone with different knowledge and skills to ensure project success. The whole-team approach means that the business representatives, testers, and developers work together in every step of the development process, from planning to delivery. The whole-team approach aims to enhance communication and collaboration within the team, leverage the various skill sets of the team members, and make quality everyone's responsibility¹². Therefore, the statement C is the most direct advantage of the whole-team approach, as it captures the essence of the principle and its benefits. The other statements are not directly related to the whole-team approach, or are incorrect. Statement A is about continuous integration, which is a practice of agile development that involves having at least once a day an automated build and test process that detects integration errors early and quickly. Continuous integration is not a direct consequence of the whole-team approach, although it may be facilitated by it¹³. Statement B is about avoiding requirements misunderstandings, which may be a benefit of the whole-team approach, but not the most direct one. The whole-team approach does not only focus on requirements, but also on design, implementation, testing, and delivery. Moreover, avoiding requirements misunderstandings may also depend on other factors, such as the quality of the user stories, the use of acceptance criteria, and the feedback from the customers and users¹⁴. Statement D is incorrect, as it contradicts the whole-team approach. The whole-team approach does not reduce the involvement of business representatives, but rather increases it. Business representatives are an integral part of the whole-team approach, as they provide the vision, the value, and the validation of the product. They collaborate with the testers and developers to define the features, prioritize the backlog, and verify the outcomes¹². References: ISTQB Foundation Level Agile Tester Syllabus¹, Section 1.2.1, page 9; What is Whole Team Approach in Agile Testing?², Section What is Whole Team Approach?; Continuous Integration³, Section What is Continuous Integration?; Effective User Stories - 3C's and INVEST Guide⁴, Section The 3 C's (Card, Conversation, Confirmation) of User Stories.

NEW QUESTION 2

Which of the following is a benefit of a whole-team approach?

- A. Enables reduction in test duplication as the testers and test engineers function as a single team.
- B. Enables testers to execute their responsibility for quality for the combined team.
- C. Enables the team to focus on solely their separate areas of expertise and responsibility.
- D. Enables the various skill sets within the team to be leveraged to the benefit of the project.

Answer: D

Explanation:

A whole-team approach is a way of working in agile projects where all team members share the responsibility for delivering high-quality software that meets the customer's needs. This means that testers, developers, business analysts, and other roles collaborate closely throughout the project lifecycle, and use their different skills and perspectives to contribute to the quality of the product. A benefit of this approach is that it enables the team to leverage the various skill sets within the team to the benefit of the project, such as testing skills, coding skills, domain knowledge, user experience, etc. This can lead to faster feedback, better communication, higher productivity, and more innovation. References: ISTQB® Foundation Level Agile Tester Syllabus¹, Section 1.1.1, page 7; ASTQB Agile Tester Certification Resources, Section 1.1.1, page 7.

NEW QUESTION 3

What is the main benefit of the Test Pyramid?

- A. It means testing is involved early in the development cycle.
- B. It helps in evaluating the amount of test cases needed.
- C. It shows complexity of testing activities.
- D. It acts as a metric for testing progress.

Answer: B

Explanation:

The Test Pyramid is a model for organizing tests in a way to make the process of testing faster, efficient and cost-effective. This model focusses on getting maximum functional testing getting covered by faster and less brittle tests like Unit and API tests¹. The main benefit of the Test Pyramid is that it helps in evaluating the amount of test cases needed for each level of testing. The Test Pyramid suggests that the number of test cases should decrease as we move up the pyramid, from unit tests to integration tests to end-to-end tests. This is because unit tests are more granular, isolated, and easy to write and maintain, while end-to-end tests are more complex, dependent, and brittle. The Test Pyramid also helps in balancing the test coverage and the test execution time, as unit tests provide high coverage and low execution time, while end-to-end tests provide low coverage and high execution time. By following the Test Pyramid, teams can optimize their testing efforts and resources, and ensure that they have a sufficient and effective test suite for their software. References: ISTQB® Foundation Level Agile Tester Syllabus, Section 2.2.1, page 16; ASTQB Agile Tester Certification Resources, Section 2.2.1, page 16; What is Test Pyramid : Getting started with Test Automation Pyramid, The Practical Test Pyramid - Martin Fowler, Testing Pyramid: What Is It and How To Use It | Solvd.

NEW QUESTION 4

Which ONE of the following is an example of a typical "Business-oriented work product"?

- A. The released product.
- B. Acceptance testing entry criteria.
- C. A user manual.
- D. Usability testing test results.

Answer: C

Explanation:

Business-oriented work products are those that describe what is needed (e.g., requirements specifications) and how to use it (e.g., user documentation). A user

manual is an example of a business-oriented work product, as it provides instructions and guidance on how to use the product from the user's perspective. A user manual may also contain information about the product's features, benefits, and limitations. A user manual is typically written by technical writers, who may collaborate with developers, testers, and business analysts to ensure the accuracy and clarity of the content. A user manual may be delivered in various formats, such as printed, online, or interactive. References: ISTQB® Foundation Level Agile Tester Syllabus1, Section 1.2.1, page 10; ASTQB Agile Tester Certification Resources2, Section 1.2.1, page 10.

NEW QUESTION 5

Which tasks are typically performed by a tester on an Agile project?

- 1) Implementing test strategy.
- 2) Documenting business requirements.
- 3) Measuring and reporting test coverage.
- 4) Coaching development team in relevant aspects of testing.
- 5) Executing test-driven development tests.

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

Answer: C

Explanation:

A tester on an Agile project typically performs the following tasks12:

? Implementing test strategy: A tester helps to define and implement the test strategy for the Agile project, which includes the test approach, test levels, test types, test techniques, test tools, test environment, test data, test metrics, and test documentation.

? Measuring and reporting test coverage: A tester measures and reports the test coverage of the product features and quality attributes, such as functionality, usability, performance, security, etc. Test coverage can be expressed in terms of test cases, test scenarios, test sessions, test conditions, test data, code, etc.

? Coaching development team in relevant aspects of testing: A tester coaches the development team in relevant aspects of testing, such as test design, test execution, test automation, test-driven development, behavior-driven development, exploratory testing, etc. A tester also helps the development team to improve their testing skills and practices.

The following tasks are not typically performed by a tester on an Agile project:

? Documenting business requirements: Business requirements are usually documented by the product owner or the business analyst, not by the tester. The tester may review and provide feedback on the business requirements, but the tester is not responsible for documenting them.

? Executing test-driven development tests: Test-driven development tests are usually executed by the developers, not by the tester. The tester may assist the developers in creating and reviewing the test-driven development tests, but the tester is not responsible for executing them.

Therefore, the correct answer is C, as it contains the tasks that are typically performed by a tester on an Agile project. References: ISTQB Foundation Level Agile Tester Extension Syllabus1, pages 14-15, 18-19, 22-23; ISTQB Agile Tester Sample Exam2, question 17.

NEW QUESTION 6

Your agile team is using the Testing Quadrants to ensure that all important test levels and test types are covered in the test plan.

In relation to Quadrant 3 - business facing and product critique, what should be considered for the plan?

- A. Exploratory Testing
B. Prototype Testing
C. Performance Testing
D. Functional Testing

Answer: A

Explanation:

Exploratory testing is a type of testing that involves simultaneous learning, test design, and test execution. It is suitable for Quadrant 3 because it is business facing

and product critique, meaning that it focuses on the user's perspective and the quality attributes of the product. Exploratory testing can help discover new risks, requirements, and defects that may not be covered by other test levels and test types. It can also provide feedback on the usability, functionality, and reliability of the product. References: ISTQB® Foundation Level Agile Tester Syllabus1, Section 2.3.2, page 17; ISTQB® Glossary of Testing Terms2, version 4.0, page 23.

NEW QUESTION 7

You are a tester in an agile team. The user story you are due to test is still under development so your tests are blocked. The main issue holding progress on this user story is that the developer's unit tests are constantly failing.

As an agile tester, which of the following actions should you take?

- A. Review the design of the problematic user story and improve it where possible.
B. Create a bug report for each of your blocked tests.
C. Work together with the developer, suggesting reasons why the tests are failing.
D. Use the time to improve and automate existing test cases of other user stories.

Answer: C

Explanation:

As an agile tester, you should work together with the developer, suggesting reasons why the tests are failing. This is an example of the agile principle of collaboration and communication within the team, as well as the agile testing practice of early and frequent feedback. By working together with the developer, you can help to identify and resolve the root causes of the test failures, as well as share your testing knowledge and perspective. This can lead to faster and better quality delivery of the user story, as well as improved team relationships and trust.

Option A is not a good action, because reviewing and improving the design of the user story is not the tester's responsibility, and it may not address the test failures. Option B is also not a good action, because creating bug reports for blocked tests is not an agile way of handling issues, and it may create unnecessary overhead and waste. Option D is not a good action, because it does not help to unblock the current user story, and it may distract you from the sprint goal and the team's focus.

References: ISTQB Foundation Level Agile Tester Syllabus, Section 2.3.1, page 171; ISTQB Foundation Level Agile Tester Sample Exam Questions, Question 2.3.1-2, page 82

NEW QUESTION 8

Which statement about an Agile task board is CORRECT?

- A. It provides detailed visual representation of the whole team's status.
- B. It is updated once at the end of each iteration.
- C. Only “in progress” tasks are shown on the task board.
- D. It is a detailed visual representation of the status of testing.

Answer: A

Explanation:

An Agile task board is a visual framework to display and sync up on the tasks moving between production steps. It is usually applied to the two most popular Agile development frameworks — Kanban and Scrum. Used by software developers and project managers, an Agile board helps manage workloads in a flexible, transparent, and iterative way¹. An Agile task board provides a detailed visual representation of the whole team's status, showing which tasks remain to be started, which are in progress, and which are done. It also helps to track the progress of the current sprint, identify bottlenecks, and facilitate collaboration and communication among team members². References:

? : ISTQB® Foundation Level Agile Tester Syllabus, Version 2014, Section 2.1.1

? : ASTQB Agile Tester Certification Resources, Agile Testing Foundations, Chapter 2, Section 2.1.1

? : 6

NEW QUESTION 9

Which of the following sentences about the integration of development and testing activities in Agile projects is INCORRECT?

- A. While developers develop automated unit test scripts, testers write automated system level tests.
- B. Testers replace developers in writing unit test automation scripts.
- C. Developers write acceptance criteria and test cases, together with testers.
- D. Developers and testers may work as a pair to develop and test a feature.

Answer: B

Explanation:

Testers replace developers in writing unit test automation scripts. Comprehensive Explanation: The integration of development and testing activities in Agile projects is based on the principle of cross-functional teamwork, where all team members collaborate and share their skills and knowledge to achieve a common goal. In the context of testing, this means that testing is not seen as a separate activity or phase, but as an integral part of the development process. Therefore, the following sentences are correct:

? While developers develop automated unit test scripts, testers write automated system level tests. This is an example of how developers and testers can work in parallel and complement each other's testing efforts. Developers can focus on testing the internal quality of the code, while testers can focus on testing the external quality of the product.

? Developers write acceptance criteria and test cases, together with testers. This is an example of how developers and testers can work together to define and verify the user requirements and expectations. Developers can provide their technical expertise and input, while testers can provide their business and user perspective and feedback.

? Developers and testers may work as a pair to develop and test a feature. This is an example of how developers and testers can work closely and interactively to deliver a feature. Developers and testers can exchange ideas, suggestions, and information, and support each other in the coding and testing tasks.

The following sentence is incorrect:

? Testers replace developers in writing unit test automation scripts. This is not a valid example of the integration of development and testing activities in Agile projects, because it implies that testers take over the responsibility of developers, rather than collaborate with them. Testers should not replace developers in writing unit test automation scripts, because developers have more knowledge and experience in coding and debugging, and because unit testing is an essential part of the development process. Testers should instead work with developers to ensure that the unit test automation scripts are adequate, effective, and maintainable. References: ISTQB® Foundation Level Agile Tester Syllabus¹, Section 1.2.1, page 9; ISTQB® Glossary of Testing Terms², version 4.0, pages 16 and 55.

NEW QUESTION 10

Which of the following is NOT a typical task performed by the tester within an Agile team?

- A. Ensuring all project status meetings are held according to the plan.
- B. Ensuring the appropriate testing tasks are scheduled during iteration planning.
- C. Suggesting improvements in team retrospectives.
- D. Working with business stakeholders to clarify requirements.

Answer: A

Explanation:

The tester within an Agile team is not responsible for ensuring all project status meetings are held according to the plan. This is typically a task for the Scrum Master, who facilitates the meetings and ensures that the team follows the Agile principles and practices. The tester within an Agile team is responsible for ensuring the appropriate testing tasks are scheduled during iteration planning, suggesting improvements in team retrospectives, and working with business stakeholders to clarify requirements. These are all tasks that contribute to the quality of the software and the testing process, as well as the collaboration and communication within the team and with the customers. References:

ISTQB Foundation Level Agile Tester Syllabus, Section 2.3.1, page 171; ISTQB Foundation Level Agile Tester Sample Exam Questions, Question 2.3.1-1, page 82

NEW QUESTION 10

Which of the following is a risk that continuous integration introduces?

- A. Teams sometimes over-rely on unit tests and exclude some important system and acceptance tests.
- B. Testers sometimes have too many builds to test, which reduces the quality of testing.
- C. Teams no longer have the ability to run manual tests, as all tests must be automated.
- D. Developer's workload is increased, which can result in a reduction of output.

Answer: B

Explanation:

Continuous integration is a practice of integrating code changes frequently and automatically into a shared repository, and running automated tests to verify the integration. Continuous integration can introduce some risks to the testing process, such as:

? Testers sometimes have too many builds to test, which reduces the quality of testing. This can happen when the code changes are too frequent or too large, and the testers do not have enough time or resources to test each build thoroughly. This can lead to missed defects, incomplete test coverage, and reduced confidence in the product quality.

? Testers sometimes have to deal with unstable or broken builds, which affects the testability of the product. This can happen when the code changes introduce errors or conflicts that cause the build to fail or malfunction. This can waste the testers' time and effort, and delay the feedback cycle.

? Testers sometimes have to cope with changing requirements and priorities, which affects the test planning and execution. This can happen when the stakeholders or customers provide new or modified requirements or feedback during the development cycle. This can require the testers to adapt their test strategy, test cases, and test data accordingly, and to balance the testing of new features and regression testing of existing features. References: ISTQB® Foundation Level Agile Tester Syllabus1, Section 2.2.3, page 14; ISTQB® Glossary of Testing Terms2, version 4.0, page 15.

NEW QUESTION 13

User Story: As a user I want to be able to calculate tax percentage based on amount of income.

What is the best black box test design technique for verifying the accuracy of this user story?

- A. Statement testing - test all statements in income calculation.
- B. User story testing - test that the user can enter an income amount and get a result.
- C. State transition testing - test all states of income entry.
- D. Equivalence partitioning - test with low, medium and high income.

Answer: D

Explanation:

The best black box test design technique for verifying the accuracy of this user story is equivalence partitioning. Equivalence partitioning is a technique that divides the input domain of a system into classes or groups that are expected to behave similarly. By testing one value from each class, the tester can reduce the number of test cases while still achieving good coverage. In this case, the input domain of the system is the amount of income, which can be divided into classes based on the tax percentage applied to different income ranges. For example, if the tax percentage is 10% for income below 10,000, 20% for income between 10,000 and 20,000, and 30% for income above 20,000, then the equivalence classes are: low income (<10,000), medium income (10,000-20,000), and high income (>20,000). By testing one value from each class, such as 5,000, 15,000, and 25,000, the tester can verify that the system calculates the correct tax percentage for each income range. This technique is more efficient and effective than testing all possible values of income, or testing only one value of income, or testing the states of income entry, or testing the statements in income calculation. References: ISTQB Foundation Level Agile Tester Syllabus1, Section 2.3.1, page 19; ISTQB Foundation Level Agile Tester Extension Sample Exam Questions2, Question 5, page 6.

NEW QUESTION 16

.....

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

CTFL-AT Practice Exam Features:

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

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