Free Questions for CTAL-ATT

Shared by England on 16-04-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You are testing a large e-commerce system for household goods that is being implemented using Agile methodologies. You are currently working on deriving tests for stories that are implementing the following epic:

As a customer, I want to use the e-commerce system, so that I can have my purchased goods delivered to my house.

The story you are currently working on is:

As a customer, I want to be told when my items will be delivered, so I can plan to be home.

You have been given the following charter that was proposed by another tester for testing this story:

Login as a customer, buy enough of each item to qualify for free shipping for each item, checkout, and verify that no shipping fee has been added.

What is the main flaw in this charter?

Options:

A- It focuses on the delivery company instead of the activities of the user
B- It does not cover the main functionality of the user story
C- The expected results are not defined
D- The actions of the user are not clearly stated in the charter

Answer:

B

Explanation:

Review of User Story:

The story's focus is on notifying the customer of the delivery time, not the shipping fee or free shipping qualification.

Analyzing the Charter:

The charter mentions verifying shipping fees, which is unrelated to the functionality of the user story.

It does not address delivery time notifications, which are the main purpose of the user story.

Alignment with Agile Testing Principles:

Agile testing requires charters to align closely with the story's intent and customer goals.

Conclusion:

The main flaw is that the charter does not test the core functionality of the story, making Option B correct.

Question 2

Question Type: MultipleChoice

Which statement correctly describes service virtualization''

SELECT ONE OPTION

Options:

A- It is a shareable testing service that simulates the behavior, data and performance of a system or service to which the product or component under test would normally be connected
B- It is a service which automatically deploys new versions of the software into the production environment at regular and short intervals
C- it is an integrated set of tools that will automatically compile, test and deliver into a virtual environment a new build of the product under test
D- it is a collection of mock objects such as stubs and drivers that will allow a product or component to run without some other product or component that it would normally need

Answer:

A

Explanation:

Service virtualization is a technique used in software testing where a virtualized service simulates the behavior of dependent components that are not available for testing. This allows testing to proceed without waiting for the actual components to be available or operational. It is particularly useful in Agile and continuous testing environments where components may be developed in parallel or where integration points may not yet be stable. Service virtualization helps to create a more controlled and deterministic testing environment, enabling more thorough testing of the

product under development.

Reference= The ISTQB Advanced Level Agile Technical Tester syllabus includes service virtualization as one of the key concepts to understand in the context of Agile testing123.

Question 3

Question Type: MultipleChoice

A unit test should be deterministic. Which option correctly describes the meaning of 'deterministic' as a characteristic of a unit test9

SELECT ONE OPTION

Options:

A- It should be small so that many tests can be run in a short period of time
B- Whenever it is run under the same conditions, it should produce the same results.
C- it should not depend on any other test.
D- It should only test the functionality related to it.

Answer:

B

Explanation:

A deterministic unit test is one that produces the same results every time it is run under the same conditions. This characteristic is crucial for unit tests because it ensures that the tests are reliable and that their results are repeatable.Deterministic tests help in identifying when a change in the codebase has introduced a defect, as any variation in the test outcome can be attributed to the change rather than an unpredictable test behavior12.

Reference= The ISTQB Advanced Level Agile Technical Tester syllabus and study resources emphasize the importance of deterministic behavior in unit tests as part of ensuring test reliability and effectiveness34.

Question 4

Question Type: MultipleChoice

What level of automation testing should be included in the production deployment process when continuous deployment is used?

Options:

A- Automated unit testing is sufficient
B- Integration and system testing
C- UAT and other acceptance testing
D- Regression testing is sufficient

Answer:

B

Explanation:

Continuous Deployment:

Continuous deployment involves automatically releasing code to production. To ensure reliability, integration and system-level testing must confirm that all components work together as expected.

Testing Scope:

While unit tests and regression tests are part of the pipeline, integration and system testing are critical for validating end-to-end functionality.

Conclusion:

Option B is correct as integration and system testing are necessary for production readiness in continuous deployment.

Question 5

Question Type: MultipleChoice

BDD and ATDD are most commonly used with which test approach?

Options:

A- Analytical
B- Model-based
C- Process-compliant
D- Reactive

Answer:

C

Explanation:

Why BDD and ATDD Align with Process-Compliant Testing:

Proactive Testing:

BDD and ATDD emphasize collaboration between stakeholders (business analysts, developers, testers, and product owners) to define the requirements before coding begins.

This proactive approach ensures that tests are integrated into the development process, reducing misunderstandings and errors upfront.

Structured Iterative Processes:

Both techniques fit seamlessly into Agile methodologies like Scrum and Kanban, where clear, iterative processes guide development and testing.

During sprint planning or backlog refinement, BDD/ATDD practices are used to refine user stories into concrete, executable test cases.

Specification by Example:

BDD and ATDD rely on defining concrete examples (acceptance criteria) collaboratively, which are directly converted into automated tests. This structured method underscores their alignment with process-oriented strategies.

Integration into CI/CD Pipelines:

The automated tests derived from BDD/ATDD become part of the CI/CD process, continuously validating that the system meets specified requirements. This exemplifies a process-compliant and proactive approach to quality assurance.

'Shift-Left' Testing Philosophy:

These practices embody the 'shift-left' approach, moving testing activities earlier in the lifecycle. This is characteristic of well-defined, process-driven methodologies.

Differentiating from Reactive:

While BDD and ATDD adapt to evolving requirements (a hallmark of Agile), they do so in a

structured, iterative way, not as a reaction to unforeseen changes or issues.

Reactive testing typically addresses late-stage feedback or emergent problems, which contrasts with the upfront planning and collaboration integral to BDD/ATDD.

Question 6

Question Type: MultipleChoice

Why might static analysis be used on an Agile project'

SELECT ONE OPTION

Options:

A- To implement a test-driven approach at the component level
B- To manually identify potential defects and thus avoid technical debt
C- To clarify and simplify code design without changing its behavior
D- To identify the causes of technical debt through the use of tools

Answer:

D

Explanation:

Static analysis in Agile projects is utilized to identify potential issues in code that could lead to technical debt. This process involves using tools to analyze the code without actually executing it, which helps in detecting complex code, potential bugs, and areas of the code that may require refactoring. By identifying these issues early in the development process, teams can address them promptly, thus maintaining code quality and avoiding the accumulation of technical debt that could hinder future development efforts.

Reference= The ISTQB Advanced Level Agile Technical Tester syllabus includes learning objectives that cover the understanding of static code analysis and its role in identifying defects and technical debt as part of code review processes in Agile projects12.

Question 7

Question Type: MultipleChoice

In a regression-averse test approach, what should be the focus of test automation?

Options:

A- To create an ever-growing set of regression tests
B- To implement automated unit testing to improve the overall quality
C- To provide automated checklists to help guide the testing
D- To continuously improve and update a set of tests that have been created

Answer:

D

Explanation:

Regression-Averse Test Approach:

This approach minimizes risks by ensuring tests remain relevant and effective over time. The key focus is maintaining and enhancing the regression suite to reflect changes in the software.

Continuous Improvement of Tests:

Keeping tests updated ensures the test suite is reliable and maintains high coverage. This involves adding tests for new features, removing obsolete tests, and refining existing ones.

Conclusion:

Option D is correct because it aligns with the goals of regression-averse approaches.

Question 8

Question Type: MultipleChoice

You are an experienced tester on an Agile team that is producing a new application. The product backlog has been refined sufficiently to begin development for the first release. This stories in this backlog contain a mixture of functional and non-functional requirements, as well as some technical stories that will provide elements of the application's infrastructure The developers have decided to use TDD as a deveiopment approach but automation of the acceptance tests is not yet planned.

You have been asked to recommend whether or not the team should take an atdd approach as well as TDD. what would be the BEST recommendation in these circumstances?

SELECT ONE OPTION

Options:

A- YES. use attd, but for the functional requirements only because it cannot be applied to non-functional or technical requirements.
B- YES. use ATTD for all requirements including non-functional and technical stories.
C- NO. do not use ATDD, because it will not work when the acceptance tests are not automated
D- NO. do not use atdd. because it is not compatible with the developers' TDD approach.

Answer:

B

Explanation:

Acceptance Test-Driven Development (ATDD) is a collaborative practice that involves the team members with different perspectives, such as customer, development, and testing, to define acceptance tests that serve as requirements1.ATDD can be applied to all types of requirements, including functional, non-functional, and technical stories, to ensure a shared understanding and agreement on the expected behavior of the software12.It is a valuable approach in Agile environments where continuous collaboration is essential, and it complements TDD by focusing on the behavior and requirements from the user's perspective3.While ATDD does not require automated acceptance tests, automating them can enhance the process by allowing for more frequent and consistent execution1.