Exam Questions CTFL-AT

Certified Tester Foundation Level Agile Tester

https://www.2passeasy.com/dumps/CTFL-AT/

NEW QUESTION 1
Which of the following activities are done in release planning?
1) Identifying testable user stories with acceptance criteria.
2) Elaborating the user stories into tasks.
3) Prioritizing the user stories.
4) Creating acceptance tests for the user stories.
5) Analyzing risks for each of the user stories.
6) Performing high level estimation for the release.

A. Activities 1, 4 and 6
B. Activities 2 and 4
C. Activities 2, 3 and 5
D. Activities 1, 3 and 6

Answer: D

Explanation:
Release planning is a process of defining the scope and timeline for an iterative or incremental product development project. It is used in agile or hybrid projects where a mid- to long-term planning of the product or system development or integration is required12. Release planning involves the following activities:
? Identifying testable user stories with acceptance criteria. User stories are short descriptions of the features or functionalities that the customer or user wants from the product. Acceptance criteria are the conditions that must be met for the user story to be considered done and acceptable. Identifying testable user stories with acceptance criteria helps to define the scope and quality of the release13.
? Prioritizing the user stories. User stories are prioritized based on the value they deliver to the customer or user, as well as the dependencies, risks, and costs associated with them. Prioritizing the user stories helps to determine the order and frequency of the releases13.
? Performing high level estimation for the release. High level estimation is a technique to estimate the effort, time, and resources needed to complete the user stories in the release. High level estimation can be done using various methods, such as analogy, expert judgment, planning poker, etc. Performing high level estimation for the release helps to set realistic and achievable goals and deadlines13.
Therefore, activities 1, 3 and 6 are done in release planning. Activities 2, 4 and 5 are done in iteration planning, which is a more detailed and short-term planning of the work to be done in each iteration or sprint13. References: 1: ISTQB® Foundation Level Agile Tester Syllabus, Section 2.2, Fundamental Agile Testing Principles, Practices and Processes1; 2: Agile Release Planning in Hybrid and Agile Projects4; 3: How to Create an Agile Release Plan5

NEW QUESTION 2
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 3
Consider an online application that allows registered users to pay the annual car tax based on the vehicle's engine power in kW. Given the following user story:
"As a customer I need the online application to calculate the annual car tax amount that I need to pay for my car:
* If the power of the vehicle is less than 20 kW, then the annual car tax is free
* If the power of the vehicle is more or equal than 20 kW but less or equal than 150 kW, then the annual car tax is 250 Euros
* If the power of the vehicle is more than 150 kW, then the annual car tax is 750 Euros" What is the MOST suitable use of a black-box test design technique for this user story?

A. Decision table testin
B. Test the following conditions:Conditions=registered user logged in; inserted power of the vehicle=20kW; Action=Car tax paid
C. State transition testin
D. Test the transitions between the following states: logging in, inserting the power of the vehicle, making payment, logging ou
E. Equivalence partitionin
F. Test the annual car tax value for the following partitions: [power of the vehicle<20 kW ; 20 kW power of the vehicles150 kW; power of the vehicle>150 kW]
G. Use case testing Test the following use case (Actor=registered user): Pre-condition=registered user logged in Scenario=registered user inserts the power of the vehicle, making payment and logs out Post-condition=car tax paid and registered user logged out

Answer: C

Explanation:
Equivalence partitioning is a black-box test design technique that divides the input domain of a system into classes of data from which test cases can be derived.

The idea is that if a system works correctly for a representative value from an equivalence class, it will work correctly for all values from that class, and vice versa. Equivalence partitioning reduces the number of test cases by eliminating redundant ones. For the given user story, equivalence partitioning is the most suitable technique because it can test the different outcomes of the annual car tax calculation based on the power of the vehicle, which is the main input for the system. By testing one value from each partition, the tester can verify the functionality of the system and detect any errors in the calculation logic. The other techniques are not as suitable because they do not focus on the inputdomain of the system, but rather on the conditions, transitions, or scenarios that are not directly related to the user story. References:
? : ISTQB® Foundation Level Agile Tester Syllabus, Version 2014, Section 2.2.2
? : ASTQB Agile Tester Certification Resources, Agile Testing Foundations, Chapter 3, Section 3.2.2
? : 3

NEW QUESTION 4
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 5
Which agile development approach incorporates the following practices:
* a project is divided into iterations called sprints
* each sprint results in a potentially releasable/shippable product?

A. Kanban
B. Extreme Programming
C. Continuous Integration
D. Scrum

Answer: D

Explanation:
Scrum is an agile development approach that incorporates the following practices:
? a project is divided into iterations called sprints, which are typically 2-4 weeks long
? each sprint starts with a planning meeting, where the team selects a subset of user stories from the product backlog to work on
? each sprint ends with a review meeting, where the team demonstrates the potentially releasable/shippable product increment to the stakeholders and collects feedback
? each sprint also includes a retrospective meeting, where the team reflects on the process and identifies areas for improvement123 References: 1: ISTQB® Foundation Level Agile Tester Syllabus, Section 2.1, Agile Software Development1; 2: ASTQB Agile Tester Certification Resources, Section 2.1, Agile Software Development2; 3: What is Agile? | Atlassian3

NEW QUESTION 6
You are developing the code that controls an industrial Espresso machine which will be operated by waiting staff in restaurants.
The machine is rather complicated and has lots of switches and buttons, so in the next iteration instructions will be provided to the operator on a small LCD screen.
A User Story for the Operator-Instructions module is as follows:
"As an operator of the Espresso machine, I would like to know how to steam milk, so I can add steamed milk to the coffee."
The following is a list of risks identified for this story, with assigned probability and impact.

A. Operators will not read the instructions and will try various switches and buttons until something work
B. Probability: Lo
C. Impact: Low
D. The instructions may be incorrect or appear in the wrong orde
E. Probability: Lo
F. Impact: High
G. An untrained customer will attempt to use the coffee machin
H. Probability: Hig
I. Impact:High
J. A small child may try to steam mil
K. Probability: Hig
L. Impact: Low

Answer: B

Explanation:
Risk-based testing is a technique that prioritizes testing activities based on the level of risk associated with each feature or requirement. The level of risk is usually calculated by multiplying the probability and impact of each risk. The higher the risk level, the more testing effort should be allocated to mitigate the risk. In this case, the risk level for each option is as follows: A. Risk level = Low x Low = Low B. Risk level = Low x High = Medium C. Risk level = High x High = High D. Risk level = High x Low = Medium Therefore, the highest risk level is C, followed by B and D, and then A. The User Story for the Operator-Instructions module should

be tested according to this risk order, starting with C, then B, then D, and finally A. Hence, the answer is B, as it is the second highest risk level and should be tested after C. References: ISTQB Foundation Level Agile Tester Extension Syllabus1, page 16; ISTQB Agile Tester Sample Exam2, question 18.

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 two of the following statements are CORRECT with regards to test automation on agile projects?
1) Every test developed for past iterations is kept and executed as part of a regression suite for each new release of code.
2) It would be very difficult to ensure high quality in an agile project without test automation.
3) Automated acceptance tests are run regularly as part of the continuous integration full system build.
4) Automated regression suites are only run for the final release of code.
5) In agile projects, the results from automated acceptance tests provide feedback on the overall product quality.

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

Answer: B

Explanation:
Test automation is essential for agile projects, as it enables fast and frequent feedback on the quality of the product. Without test automation, it would be very difficult to ensure high quality in an agile project, as manual testing would be too slow and costly to keep up with the pace of development12. Automated acceptance tests are one of the key types of test automation on agile projects, as they verify that the user stories are implemented correctly and that the product meets the customer's expectations. Automated acceptance tests are run regularly as part of the continuous integration full system build, and the results provide feedback on the overall product quality12. Therefore, statements 2
and 5 are correct with regards to test automation on agile projects. References: 1: ISTQB® Foundation Level Agile Tester Syllabus, Section 3.3.1, Test Automation1; 2: ASTQB Agile Tester Certification Resources, Section 3.3.1, Test Automation2

NEW QUESTION 9
Which of the following sentences related to Risk-based testing is CORRECT?

A. Risk-based testing fits well in Agile development processes, as risks are analyzed twice- during release and iteration planning.
B. Risk-based testing does not fit well in Agile development processes, as short iterations mandate short test times.
C. Risk-based testing does not fit in Agile development processes, as each iteration focuses on limited parts of the product.
D. Risk-based testing fits well in Agile development processes, as risks are easy to identify when the work is divided into user stories.

Answer: D

Explanation:
Risk-based testing fits well in Agile development processes, as risks are easy to identify when the work is divided into user stories. User stories are short descriptions of features or functionalities that are valuable to the customers or users. They help to define the scope and priority of the work in each iteration. By breaking down the work into user stories, the Agile team can identify the potential risks associated with each story, such as technical complexity, business criticality, or user feedback. The team can then prioritize the testing effort based on the risk level of each story, ensuring that the most important and risky features are tested first and thoroughly. Risk-based testing also helps to optimize the testing time and resources, as the team can focus on testing the most relevant aspects of the software, rather than testing everything equally. References: ISTQB Foundation Level Agile Tester Syllabus, Section 2.3.2, page 181; ISTQB Foundation Level Agile Tester Sample Exam Questions, Question 2.3.2-1, page 92

NEW QUESTION 10
.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual CTFL-AT Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the CTFL-AT Product From:

https://www.2passeasy.com/dumps/CTFL-AT/

Money Back Guarantee

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