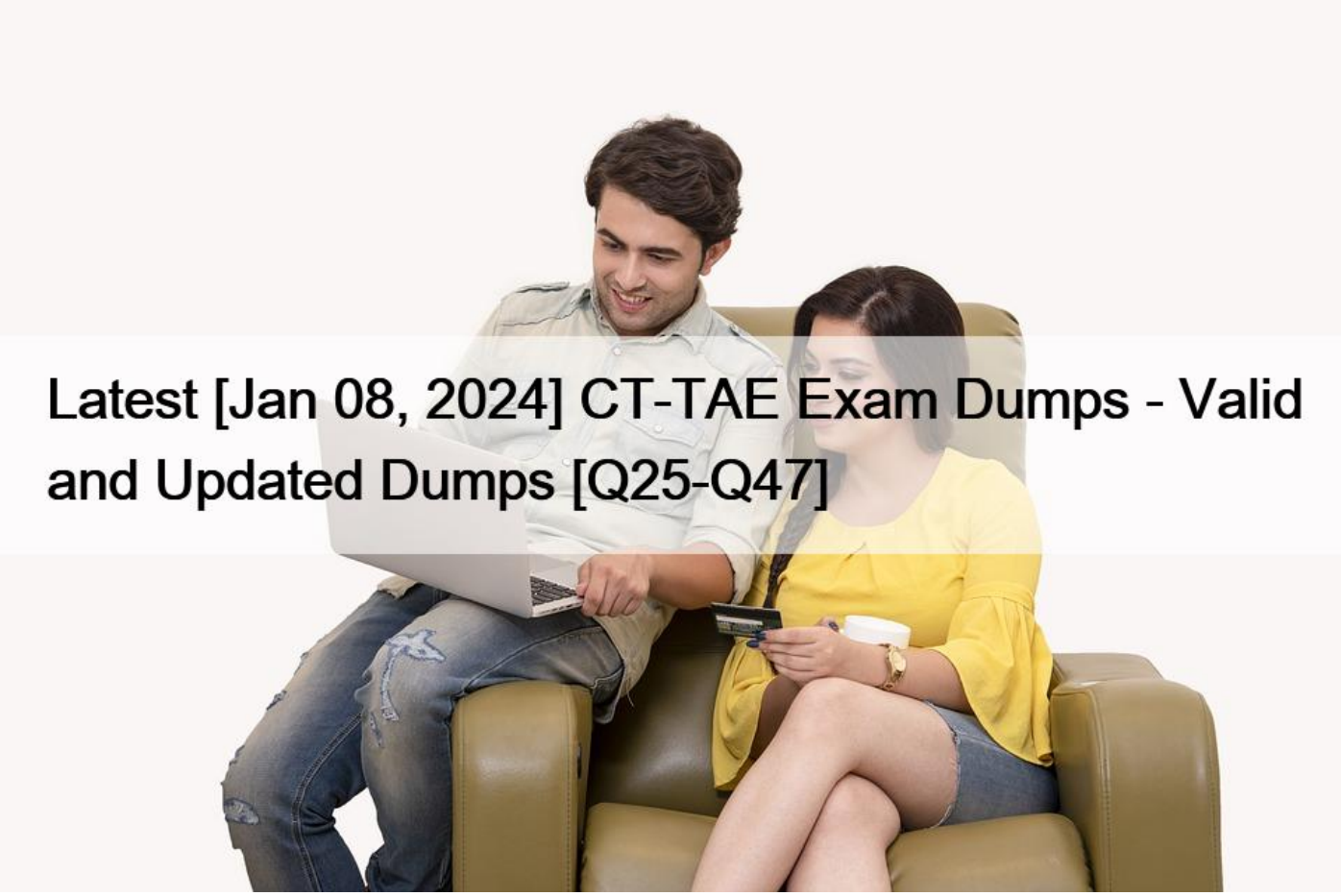
Latest [Jan 08, 2024 CT-TAE Exam Dumps - Valid and Updated Dumps [Q25-Q47



Latest [Jan 08, 2024] CT-TAE Exam Dumps - Valid and Updated Dumps
Free Sales Ending Soon - 100% Valid CT-TAE Exam Dumps with 82 Questions

ISTQB CT-TAE (Certified Tester Test Automation Engineer) Exam is a globally recognized certification for software testers who specialize in test automation. Certified Tester Test Automation Engineer certification is designed to validate the knowledge and skills of professionals who work in test automation, including test automation engineers, software developers, and quality assurance (QA) professionals. Certified Tester Test Automation Engineer certification covers a wide range of topics related to test automation, including test automation design and development, test automation architecture, and test automation management.

ISTQB CT-TAE (Certified Tester Test Automation Engineer) certification exam is a professional qualification that is designed to equip software testers with skills and knowledge in test automation. Certified Tester Test Automation Engineer certification is ideal for individuals who are interested in pursuing a career in software testing, particularly in test automation. Certified Tester Test Automation Engineer certification is globally recognized, and it is a valuable addition to the resume of any software tester.

Q25. You have been asked to determine a TAS for a new release of a SUT, test should be automated wherever. The new release will consist of 5 new interfaces and an amendment to 3 existing interfaces. The new and amended interface will be deliver incrementally in 3 sprints, each lasting 2 weeks.

What would be the BEST Test Automation Solution (TAS) design in this scenario?
* Automate tests at both Component and System Level. Only do this automation once every interface has been fully developed or amended and manual testing has completed successfully.
* Automate tests at one level only, System level. Use only the newly developed interfaces and do not create any customized interfaces/test hooks.
* Automate the tests at two levels, Component and System level. Create customized hooks at Component level for interface not yet developed or amended. Only use the newly developed or amended interfaces to test at System level.
* Automate a test at once level, component level, Create customized interface/test hooks for this level where the interface has not yet been developed or amended.

Q26. A major component of your organisation’s Test Automaton Solution (TAS) is a popular open-source third-party capture-replay tool for automated functional testing.

Which two of the following must the Test Automation Engineer (TAE) ensure happens for this TAS?

a) The third party tool is placed under configuration management control.

b) The annual support and maintenance costs are agreed with the tool’s vendor.

c) It is Important to obtain information about updates and new versions of the tool so that the third party tool is kept up to date.

d) Ensure that the TAS test scripts are integrated into the tool’s framework.

e) Ensure that no changes are made to the tool, because modifications are not allowed for third party products.
* a and b
* c and d
* a and c
* d and e

Q27. A regression test suite consist of 500 test cases which are all executed manually. The business case for a pilot project is based on the adoption of test automation using a commercial tool that will reduce the execution time by a factor of 90% for 100% of the tests in the regression test suite. The pilot project lasted one month (as planned) and you are currently its results. At the end of the pilot project, 40% of the regression tests have been automated and their execution time has been reduce by 60%.

Which of the following statements is TRUE in this scenario?
* The duration of the pilot project was too short -it should last unit the success factors are achieved
* The target defined for the business case is too accurate -it should not be measureable
* The project selected for the pilot is too critical -if should not be too critical or too trivial
* The target defined for the business case seems difficult to hit – it should be realistic

Q28. Consider a TAS that is going to be deployed for the first time. The TAS requires share resources and run it its own test environment. The infrastructure for the TAS has been created along with maintenance procedures. It is very unlikely the TAS will be required to work in other target Environments. There is a high-risk that when the TAS is deployed in its own test environment, a number of existing application will no longer work because of conflicts with the existing shared resources.

Which of the following activities would you expect to be MOST effective at mitigating the risk associated with the first deployment

of the TAS?
* Testing the TAS for application compatibility issues in the target environment
* Testing the TAS for its ability to be implemented in other target test environments.
* Testing the TAS for regressions due to optimization that fix non-functional issues.
* Testing the TAS for ITS ability to run a shared test environment

Q29. Designing the System Under Test (SUT) for testability is important for a good test automation approach and can also benefit manual test execution.

Which of the following is NOT a consideration when designing for testability?
* Observability: The SUT needs to provide interface that give insight into the system.
* Re-useability: The code written for the SUT must be re-useable for other similar system.
* Clearly defined architecture: The SUT Architecture needs to provide clear and understandable interfaces giving control and visibility on all test levels.
* Control: the SUT needs to provide interfaces that can be used to perform actions on SUT.

Q30. Consider a SUT that small run on multiple platform during the execution of automated test runs. In each test run an automated test suite needs to be executed, with the same version of the TAF, against the same version of the SUT of each platform. Each platform shall have its own dedicated test environment. Your goal is to implement a process as automated as possible (i.e with minimal manual intervention) that allows implementing a consistent setup of the TAS across the multiple test environments.

Which two of the following aspects are MOST relevant for achieving your goal in this scenario?

The configuration of the TAS uses automated installation scripts

The TAF saves the logs needed to debug errors in XML format

C) Features of the TAF not used by the automated tests have been tested D) All the automated test cases contain the expected results E) The TAS components are under configuration management
* A and e
* B and c
* B and d
* A and d

Q31. You are planning the pilot for an in-house developed Test Automation solution (TAS).

Which two of the following would be important steps to take as part of the planning process?

a) Review your organisation’s current projects and identify which one would be most suitable to pilot the TAS.

b) Ensure that the developers will provide the necessary commitment for the TAS deployment activities.

c) Run a series of training workshops for new users of the TAS before they are asked to use it.

d) Develop a project plan for the pilot and reserve the necessary budget and resources for its implementation.

e) Ask the developers to provide any missing functionality during the deployment activities.
* a and b
* b and d
* c and d

* c and e

Q32. Which of the following CORRECTLY describes how automation SHOULD be applied to confirmation testing?
* Confirmation tests are not good candidates for automation as they are not designed to run many times
* Confirmation tests should only be automated if they fail to pass on the first attempt
* Confirmation tests can be automated and incorporated into an automated regression suite to show whether defects that were previously fixed reoccur
* A confirmation test should only be automated after it has been run manually

Q33. You are working on a TAS for standalone application. The automated tests are developed based on a automation framework that allows interaction with GUI elements using on object orientated API. The GUI elements include menus, buttons, radio buttons, text toolbars and their properties.

Whilst automating a test, you have discovered that the GUI elements of some third party components are not identifiable by the automated tool you are using.

Which of the following is the FIRST step that you take to investigate this issue?
* Verify the testability support with the providers of the third party components
* Verify whether the GUI identification depends on the browser.
* Adopt an approach that uses the coordinates of the GUI elements instead
* Verify whether naming standards for variables and have been defined for the current automation solution

Q34. Consider a TAS deployed into production. The SUT is a web application and the test suite consists of a set of automated regression tests developed via GUI. A keyword-driven framework has been adopted for automating the regression tests. The tests are based on identification at low-levels of the web page components (e.g class indexes, tab sequence indexes and coordinates) in the next planned release the SUT will be subject to significant corrective maintenance (bug-fixes) and evolution (new features) Maintenance costs to update the test scripts should be as low as possible and the scripts must be highly reusable.

Which of the following statements is most likely to be TRUE?
* The keyword-driven framework is not suitable, it would be better to adopt a structured-scripting approach
* False positive errors are likely to occur when running the automated tests on the new releases without modifying the test
* The total execution time of the automated regression test suite will decrease for each planned release.
* The keyword-driven framework introduces a level abstraction that is too high and makes it difficult what really happens

Q35. Consider a TAS that uses a keyword-driven framework. The SUT is a web application and there is a large set of keywords available for writing the automated tests that relate to highly specific user actions linked directly to the GUI of the SUT. The automated test written with the keywords are statically analyzed by a custom tool which highlight’s repeated instances of identical sequence of keywords. The waiting mechanism implemented by the TAS for a webpage load is based on a synchronous sampling within a given timeout. The TAS allows checking a webpage load every seconds until a timeout value
* Changing the scripting approach to data-driven scripting
* Implementing keywords with a higher level of granularity
* Changing the wait mechanism to explicit hard-coded waits
* Establishing an error recovery process for TAS and SUT

Q36. Consider the following layers of the gTAA structure:

a. Test generation layer

b. Test definition layer

c. Test execution layer

d. Test execution layer

Consider the following capabilities associated with these layers.

Acquire all the necessary resources before each test and release all after run, in order to avoid interdependences between test Allow the automated test scripts on an abstract level to interact with components, configurations and interfaces of the SUT.

Design test directives that allow configuring the algorithms used to automatically produce the test cases a given model of the SUT.

Allow the definition and implementation of test cases and data by means of templates and/or guidelines.

Which of the following BEST matches each layer with the appropriate capability?
* a-3, b-4, c-1, d-2
* a-4, b-3, c-1, d-2
* a-4, b-3, c-2, d-1
* a-3, b-4, c-2, d-1

Q37. A web application was released into production one year ago, it has regular release which follow a V-model lifecycle and testing is well-established and fully integration into the development lifecycle. You have been asked to implement a TAS for the regression test suite. The regression tests have been developed via the GUI and are expected to be run at least four times a month, for each planned release, for the whole operation solution life of the system (six years). Each screen of the GUI uses several third-party controls which are not compatible with the existing automation solutions. The environment for the automation will be stable, fully controllable and separated from other environments (development, staging, production).

What could be the MOST problematic for this TAS?
* Maturity of the test process
* Complexity to automate
* Frequency of use
* Sustainability of the automated environment

Q38. You are using a gTAA to create a TAS for a project. The TAS is aimed specifically at automating a suit of existing manual test cases for standalone desktop applications. All the interfaces between the TAS and SUT will be from the CUI of the application.

Which of the following layers of the gTAA should you focus on for the TAS?
* The test Generation layer
* The Test Definition layer
* The Test Adaption layer
* The Test Execution layer

Q39. You are using a gTAA to create a TAS for a project. The TAS is aimed at automatically and executing test cases based on a use-case Modeling approaching that uses UML as a modeling language. All the interaction between TAS and SUT will only be at the API and GUI level.

Which of the following components of the gTAA would you EXCLUDE from the TAS?
* The test reporting component of the test execution layer.
* The Test execution component of the test generation layer
* The test execution (test engine of the test execution layer
* The Command Line Interface (CLI) component of the test adaptation layer

Q40. You are implementing test automation for a project and you want to be able to generate test cases automatically using a series of test design tools which use a variety of test design techniques such as decision tables, pairwise testing and boundary value analysis.

You also want to generate test data automatically which can then be used by the tests.

Initially these tests will be run manually to verify their correctness and ultimately you want to include them in the test execution tool so that they can run unattended.

Which layer of the gTAA will be used to support the specification of the test cases and preparation of the test data?
* The generation layer
* The definition layer
* The execution layer
* The adaptation layer

Q41. You have implemented a keyword-driven scripting framework, which uses a test execution tool to run the tests. This has been in use for the past year and all of the teams now use this framework as the standard approach for test execution.

The teams all work on different aspects of the SUT and they have all experienced significant benefits in the use of this scripting framework. However, on closer examination, you have discovered that there are numerous instances where the teams have the same functionality to test but are using different keywords.

One of your objectives for improvement is to create consistency among the teams.

What is the BEST way to handle this situation?
* Move to a model-based approach to scripting where the models include the keywords.
* Do nothing, each team are working in isolation and they are all experiencing significant benefits in the way they are currently working.
* Provide each team with a set of guidelines and naming conventions for keywords.
* Create a central library of keywords and associated definitions for each team to use.

Q42. You identified a suitable project to pilot an automation tool and planned and conduced a pilot. The pilot has been successful and tool Is being deployed within your organization, with a plan to increase tool use by the one project at a time. During this rollout some test processes will be changed slightly to gain additional benefits from using the tool.

In the pilot project, a small set of manual tests were automated for the first time. You are currently monitoring the test automation efficiency and this reveals that the automation regime for the tests is not yet mature.

Which of the following statements is TRUE?
* The approach used for deployed this tool is aligned to the standard success factor for deployment
* The pilot project should have been critical so that maximum benefits were delivered
* The target defined for the project was inappropriate, because the automation regime for the automated tests at the end of the pilot is not yet mature.
* The test process should be radically changed to gain additional benefits from using the tool.

Q43. A defect in a SUT has been resolved and validated by an automated defect re-test in the current release of the software. This retest has now been added to the automated regression test suite.

Which statement BEST describes a reason why this defect could re-occur in future releases?

* Automated defect confirmation testing is not effective at confirming that the resolved defect will continue to work in future releases
* The configuration management process does not properly control the synchronization between software archives
* The automated regression test suite is not run consistently for future releases.
* The automated regression test suite has a narrower scope of functionality

Q44. Consider A TAS for testing a desktop application via its GUI. All the test cases of the automated test suite contain the same identical sequences of steps at the beginning (to create the necessary objects when doing a preliminary configuration of the test environment and at the end (to remove everything created -specifically for the test itself during the preliminary configuration of the test environment). All automated test cases use the same set of assertion functions from a shared library, for verifying the values in the GUI fields (e.g text boxes).

What is the BEST recommendation for improving the TAS?
* Implementing keywords with higher level of granularity
* Improving the architecture of the application in order to improve its testability
* Adopting a set of standard verification methods for use by all automated tests
* Implementing standard setup and teardown functions at test case level

Q45. Consider a TAS associated to dynamically changing software frequent releases. Your goal is to determine the amount of effort required to maintain the automated tests of the regression test suite for each new release of the SUT.

What is the MOST important metric to collect to achieve your goal?
* The code coverage achieved with the automated tests, for each new release of the SUT
* The number of automated tests which fail because of a single software defect, for each new release of the SUT
* The time it takes to execute all the automated tests, for each new release of the SUT.
* The number of automated tests requiring maintenance, for each new release of the SUT.

Q46. A TAS uses a commercial test automation tool and the default logs generated by the inconsistent formats such as different types of messages (pass/fail steps, screenshots, warnings, etc.) To solve this issue some custom logging functions have been created from the test scripts, making it possible to log the different types of messages with the same format. However, this may cause a problem due to excessive size of the logs which can make it difficult to find the required information. Assume that all the default logs will be disabled when running the automated tests and that some tests will not generate excessively sized logs.

Which of the following represents the BEST suggestion for implementing the custom logging functions?
* Implement the custom logging functions without saving timestamps
* Implement the custom logging functions to support different levels of tracing
* Implement the custom logging functions without saving stack traces
* Implement the custom logging functions to redirect the logs to multiple files

Q47. You are currently designing the TAA of a TAS. You have been asked to adopt an approach for automatically generating and executing test cases from a model that defines the SUT. The SUT is a state-based and event-driven that is described by a finite-state machine and exposes its functionality via an API. The behavior of the SUT depends on hardware and communication links that can be unreliable.

Which of the following aspects is MOST important when designing the TAA in this scenario?
* Looking for tools that allows direct denoting of exceptions and actions depending on the SUT events.
* Adopting a test definition strategy based on classification tree coverage for the test definition layer.
* Looking for tools that allow performing setup and teardown of the test suites and the SUT.
* Adopting a test definition strategy based on use case/exception case coverage for the definition layer.

CT-TAE Exam Dumps - 100% Marks In CT-TAE Exam: https://www.validbraindumps.com/CT-TAE-exam-prep.html]