

EXAMTOPICS

- Expert Verified, Online, **Free**.



CERTIFICATION TEST

- [CertificationTest.net](https://www.CertificationTest.net) - Cheap & Quality Resources With Best Support

You are working as a TAE for a company who have been using a web test execution tool for a number of years. The tool has been used successfully on ten web applications in the past.

The company are developing a new web application which has a friendly User Interface, but the developers have used an object throughout the application which the tool is unable to recognise. As a result, you have no way of capturing the object or verifying the contents using the automation tool.

What is the first thing you should do about this problem?

- A. See if the application can be run on a desktop and if the object can be recognised on the desktop by the tool.
- B. Investigate whether the object can be recognised by other test execution tools in the market
- C. Ask the developers to remove the object and replace it with some text fields
- D. Ask the developers if they can change the object to something that can be recognised by the tool

Suggested Answer: D

Community vote distribution

D (67%)

B (33%)

FrancisFerreira Highly Voted 2 years, 8 months ago

Selected Answer: D

At first glance, it should be irrelevant whether or not other tools in the market can interact with such object, because changing your TAS should definitely NOT be your first thought. First one should address their devs about the possibility of using standard objects that the current TAS can interact with instead.

upvoted 5 times

sozdursun Most Recent 1 month, 2 weeks ago

Selected Answer: D

In the official sample exam (v2.2 answers), this exact scenario is covered. The correct response is:

Correct Answer: D – Ask the developers if they can change the object to something that can be recognised by the tool

Why D is correct

It addresses the problem at its root: the automation tool cannot interact with the custom object.

Collaboration with developers to make the object “testable” (e.g., adding IDs, accessibility hooks, or replacing with recognizable controls) is aligned with testability principles from the syllabus.

Options A and B are “workarounds” that don’t solve the underlying maintainability/testability problem.

Option C (removing the object) is unrealistic and disruptive to the application’s design.

✓ So yes, your answer D is correct.

upvoted 1 times

Einram 8 months ago

Selected Answer: D

D. Ask the developers if they can change the object to something that can be recognised by the tool

upvoted 1 times

aasilva 1 year ago

Selected Answer: B

B is the right answer

upvoted 1 times

Kshwa 1 year, 4 months ago

B is the right answer

upvoted 1 times

🗨️ 👤 **Tarannnnnnnn** 1 year, 6 months ago

Which is the correct answer to get the right option in on the ISTQB TAE exam? please tell I have tow rite exam tomorrow

upvoted 1 times

🗨️ 👤 **AkshayTadla** 1 year, 7 months ago

Selected Answer: B

B is the right answer

upvoted 2 times

🗨️ 👤 **kavee123** 1 year, 10 months ago

D is the right answers

upvoted 1 times

🗨️ 👤 **khanhcuong** 1 year, 11 months ago

Selected Answer: B

Firstly, I think we should investigate to find our own solution, then if it does not work so we can ask the developer to help

upvoted 1 times

🗨️ 👤 **binhlex** 2 years, 3 months ago

As a TAE, we should do investigate the problem before passing it to others (devs), B might take more effort but we will have evidences of our research when talking the problem with developers.

upvoted 1 times

🗨️ 👤 **LeoTran1708** 2 years, 4 months ago

Selected Answer: D

WWe should cope with it with our inter developers in team first.

upvoted 1 times

🗨️ 👤 **anhtho** 2 years, 4 months ago

Selected Answer: D

B is a solution that may take too much effort, this question asks for the FIRST thing to try, so D is most likely to try first.

upvoted 2 times

🗨️ 👤 **Gsha** 2 years, 7 months ago

None of these questions appeared in the exam. The pattern is similar although.

upvoted 2 times

🗨️ 👤 **gbkallur** 2 years, 7 months ago

Also question 1 to 16 repeats every exam?

upvoted 1 times

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. a and b
- B. c and d
- C. a and c
- D. d and e






Suggested Answer: C


Community vote distribution

C (100%)

 **sozdursun** 1 month, 2 weeks ago

Selected Answer: C

- a) Put the tool under configuration management →  Correct. Even third-party tools must be under CM so versions are tracked and reproducibility is guaranteed.
- b) Annual support/maintenance costs →  Not relevant for open-source (no vendor contract).
- c) Keep tool updated with new versions →  Correct. Otherwise you risk incompatibility, security issues, or bugs.
- d) Ensure TAS scripts are integrated into tool's framework →  Too specific; integration is about TAS design, not mandatory for CM.
- e) Ensure no changes are made →  Not true – you can extend/wrap open-source tools if needed.

 So the correct pair is: a and c → Option C
upvoted 1 times


 **Einram** 8 months ago

Selected Answer: C

C. a and c
upvoted 1 times

 **wallace2808** 1 year ago

Are these solutions legit? The first 2 are all against my judgement.
upvoted 1 times

 **sctmp** 1 year, 1 month ago

Selected Answer: C

C is correct
upvoted 4 times

 **gbkallur** 1 year, 1 month ago

Also question 1 to 16 repeats every exam?
upvoted 2 times

 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: C

Support and maintenance costs are contractual things that don't usually concern the TAE. The TAE, however, does need to be aware of new releases and updates for the third-party tool so that no technical debts are generated for missing out on those.
upvoted 4 times

 **FrancisFerreira** 1 year, 1 month ago

Not to mention it is an open-source tool, so there is likely no support or maintenance costs to worry about.
upvoted 3 times

If model-based testing has been selected for the overall test automation approach for a project, how does that influence the layers of the TAA?

- A. All layers are used, but the test generation layer will be automated based on the defined model
- B. There will be no need for the execution layer
- C. No adaptation will be needed because the interfaces will be defined by the model
- D. There will be no need to design the tests for the API because those will be covered by the model

Suggested Answer: A

Community vote distribution

A (100%)

 **Einram** 8 months ago

Selected Answer: A

A. All layers are used, but the test generation layer will be automated based on the defined model
upvoted 1 times

 **bberries** 12 months ago

Selected Answer: A

A is the correct answer.

"Test generation: Supports the automated design of test cases based on a test model. Model-based testing tools can be leveraged in the generation process (see the ISTQB CT-MBT Syllabus). Test generation is an optional capability"

...cited from 3.1.1 - Explain the Major Capabilities in a Test Automation Architecture

upvoted 1 times

 **FrancisFerreira** 2 years, 8 months ago

Selected Answer: A

Alternative A seems correct.

upvoted 1 times

Your functional regression test automation suite ran successfully for the first two sprints and no failures were encountered during the runs. The automation suite records the status of each test case as either 'pass' or 'fail' and has excellent recovery capability built in.

For the third sprint, the TAS log reported several test cases with a status of 'fail'. You investigated each test case and found that most failures were due to a defect in one of the keyword scripts, rather than in the SUT. For those where the failure was in the SUT, defect reports were raised but several were returned by the developers asking for more information to enable them to reproduce the problem.

Which additional log items SHOULD you add to the TAS that would BEST improve failure analysis and defect reporting for future sprints? a)

Dynamic measurement information about the SUT. b) A status of 'TAS error', in addition to 'pass' and 'fail', for each test case. c) Use of a colour coding scheme so that 'pass' is in red and 'fail' is in green. d) A counter to determine how many times each test case has been executed. e)

System configuration information including software/firmware and operating system versions. f) A copy of the source code for all Keyword scripts executed.

A. a and b

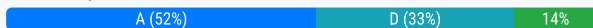
B. d and e

C. a and c

D. b and e

Suggested Answer: A

Community vote distribution



sozdursun 1 month, 2 weeks ago

Selected Answer: D

Evaluate the options

a) Dynamic measurement info about the SUT → Nice-to-have, but not essential for distinguishing TAS errors or helping reproduction.

b) Add "TAS error" status → Very useful, clearly separates framework issues from SUT bugs.

c) Colour coding (pass=red, fail=green) → Cosmetic, not useful for analysis.

d) Counter of executions → Not directly helpful for root cause analysis or defect reproduction.

e) System configuration info (OS, firmware, software versions) → Critical for devs to reproduce bugs.

f) Copy of source code for keyword scripts → Overkill, better to fix in version control, not logs.

Correct Answer

D. b and e

upvoted 1 times

K_sharma_2000 2 months ago

Selected Answer: B

Option d: A counter to determine how many times each test case has been executed is valuable for failure analysis. It provides historical execution data, which can help identify patterns (e.g., intermittent failures) and assess test case reliability over time. This aligns with Section 6.1.2 of the CTAL-TAE syllabus ("Analyze Data from the Test Automation Solution and the System Under Test to Better Understand Test Results"), which emphasizes analyzing test data to improve understanding.

Option e: System configuration information including software/firmware and operating system versions is critical for defect reporting. This data allows developers to replicate the exact environment where the SUT failure occurred, addressing the scenario where defect reports were returned due to insufficient reproduction details

upvoted 1 times

pabc08 1 year, 1 month ago

Selected Answer: A

Chapter 5.3:

- The execution status of the test case can be pass, fail or TAS error. The result of TAS error is used for situations where the problem is not in the SUT.
- Dynamic information about the SUT (e.g., memory leaks) that the test case was able to identify with the help of third party tools. Actual results and failures of these dynamic measurements should be logged with the test case that was executing when the incident was detected.

upvoted 3 times

🗨️ **issoChan** 1 year, 2 months ago

In section 5.3 it is explained. For TAS; a and b are clear. Counter is needed in case of reliability/stress testing but there is no mention in text for them. So correct answer is A

upvoted 2 times

🗨️ **Leandro__** 1 year, 2 months ago

Selected Answer: D

D. b and e

upvoted 1 times

🗨️ **Svarog** 1 year, 3 months ago

Selected Answer: D

The correct answer is:

D. b and e

Explanation:

Adding a status of 'TAS error' (option b) would provide additional information about the nature of the failure, allowing for better categorization and analysis.

Including system configuration information, such as software/firmware and operating system versions (option e), would help in understanding the environment in which the failures occurred, aiding in defect reproduction and resolution.

(s. chapter 1.2: "Enable easy troubleshooting: In addition to the test execution and logging, the TAF has to provide an easy way to troubleshoot failing tests. The test can fail due to

o failures found in the SUT

o failures found in the TAS

o problem with the tests themselves or the test environment."

upvoted 2 times

🗨️ **khanhcuong** 1 year, 5 months ago

Selected Answer: A

when debugging automation script, we need more information than just passed or failed, that the reason why we need a) Dynamic measurement information about the SUT and b) A status of TAS error', in addition to pass' and 'fail', for each test case.

upvoted 2 times

🗨️ **Tisaja** 1 year, 7 months ago

Selected Answer: B

System configuration information including software/firmware and operating system versions (e) is a must have in this situation. Therefore we need to choose between B and D. Log item "b" is unrealistic and log item "d" would be helpful as we can see if the test has been run only once or it was executed also in the rerun. That leaves us with the correct answer "B".

upvoted 3 times

🗨️ **anhtho** 1 year, 10 months ago

Selected Answer: A

Answer A is the correct answer because both the 'TAS error' and the 'Dynamic measurement information about the SUT' are listed as TAS log requirements. Answer D is incorrect because System configuration is not a TAS log requirement but a SUT log requirement.

upvoted 3 times

🗨️ **es12398080** 1 year, 10 months ago

Selected Answer: A

Answer A is the correct answer, because both the 'TAS error' and the 'Dynamic measurement information about the SUT' are listed as TAS log requirements. Answer D is incorrect, because System configuration is not a TAS log requirement but a SUT log requirement.

upvoted 4 times

🗨️ **wallace2808** 2 years ago

Again, I have to agree with community answer. D is most logical.

upvoted 2 times

  **gbkallur** 2 years, 1 month ago

Also question 1 to 16 repeats every exam? How come?

upvoted 1 times

  **FrancisFerreira** 2 years, 1 month ago

Selected Answer: D

An execution counter would be of no help whatsoever. However, the ability to recognize TAS failures and report those would save lots of time during defect investigation.

upvoted 4 times

Your TAS has been running successfully on a Windows/GUI based SUT for some years. The SUT has undergone minimal change over the years to maintain business as usual, deploying six-monthly releases for minor enhancements and bug fixes using a waterfall lifecycle. The TAS has not changed at all during this period. The current project for the SUT will be using the Scrum methodology to deliver a more modern, competitive, user interface. It is in the release planning stage with an agreed release backlog and set of sprints outlined. The move from lengthy waterfall releases to shorter sprints has led you to conduct a review of the current TAS to make sure it is robust and fully optimised for the timescale challenges of the new project.

What two steps would be BEST to undertake during the review?

a) Ensure that new automation code is using the same naming conventions as existing code. b) Perform a full regression run in Sprint 1 to identify what improvements could be made to the TAS for future sprints. c) Ensure that the TAS is using the latest libraries for the operating system. d) Review the functions that act upon the controls for the GUI for possible consolidation. e) Involve the test team to see what ease-of-use improvements they would like to see made to the TAS.

- A. c and d
- B. b and c
- C. a and b
- D. d and e

Suggested Answer: B

Community vote distribution



sozdursun 1 month, 2 weeks ago

Selected Answer: A

Option Analysis

- a) Ensure naming conventions match existing code → ❌ Minor cosmetic improvement, not critical for sprint readiness.
- b) Perform full regression in Sprint 1 → ❌ Not efficient. In Scrum, you want lightweight regression (smoke/sanity) each sprint, not a huge upfront regression that delays progress.
- c) Ensure TAS uses latest OS libraries → ✅ Important. Outdated libraries can block stability, performance, and compatibility. Keeping TAS updated reduces risk during fast-paced sprints.
- d) Review GUI functions for consolidation → ✅ Yes. Refactoring redundant/duplicated functions improves maintainability and speed – crucial when sprint cycles are short.
- e) Involve test team for ease-of-use feedback → Nice-to-have, but secondary compared to technical robustness and maintainability.

✅ Correct Answer

A. c and d
upvoted 1 times

Zinayozgat 7 months, 2 weeks ago

Selected Answer: D

Since The TAS has remained unchanged while SUT is shifting to a Scrum-based release cycle, it is important to ensure the TAS remains efficient and adaptable.

D- helps ensure that redundants or inefficient automation code streamline

E- better maintainability and ease of use

.
upvoted 1 times

Sbmk 7 months, 3 weeks ago

Selected Answer: D

- (d) Review the functions that act upon the controls for the GUI for possible consolidation
- The UI is being modernized, meaning GUI elements may change frequently. Reviewing and consolidating functions will improve maintainability and reduce duplication in test automation.
- (e) Involve the test team to see what ease-of-use improvements they would like to see made to the TAS
- Since Agile involves continuous testing and collaboration, gathering feedback from testers ensures the TAS is efficient and easy to use in the new iterative development cycle.

upvoted 2 times

🗨️ 👤 **Wokawokaeee** 1 year, 4 months ago

Selected Answer: A

why it is not A? Full regression on first sprint can lead to not actual/important improvements. as per scrum should be give effort on most valuable projects/item/features?

upvoted 2 times

🗨️ 👤 **FrancisFerreira** 2 years, 1 month ago

Selected Answer: B

Alternative B seems correct.

upvoted 2 times

You are working on a government system called "Making Tax Digital" or MTD for short. This system is being implemented to stop manual human input error and also to reduce fraudulent behaviour from companies when submitting their tax and VAT returns. The key concept is that registered companies will need to use government recommended 3rd party software for their accounts and book keeping. These 3rd party applications will have a direct interface into the government's main system for transactions and submissions. You have been using a test execution tool successfully on the project so far. and have implemented a basic "capture/replay" approach to scripting.

The management have been encouraged with the automation so far, but want the following objectives to be met:

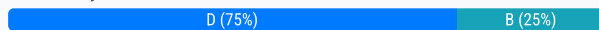
- Test cases added easily
- Reduction in the amount of scripts and script duplication
- Reduction in maintenance costs

Which scripting technique would be MOST suitable in this scenario in order to meet the objectives?

- A. Linear scripting
- B. Structured scripting
- C. Data-driven scripting
- D. Keyword-driven scripting

Suggested Answer: D

Community vote distribution



sozdursun 1 month, 2 weeks ago

Selected Answer: D

ption Analysis

- A. Linear scripting → ❌ This is what capture/replay already does → lots of duplication, high maintenance.
- B. Structured scripting → ❌ Better than linear (reusable functions), but still limited in flexibility and harder to scale.
- C. Data-driven scripting → ✅ Good for reducing duplication when the same flow is tested with many input variations. But here, multiple different types of test cases need to be added. Data-driven alone won't solve all maintainability and duplication problems.
- D. Keyword-driven scripting → ✅ Best choice. Keywords abstract actions into reusable units.

Test cases can be added easily by combining keywords.

Reduces duplication since keywords are reused.

Reduces maintenance – only update the keyword implementation, not every test.

✅ Correct Answer

D. Keyword-driven scripting
upvoted 1 times

nemeremjcc 10 months ago

Selected Answer: C

Data driven is easy to add test cases and can reduce coast as well as duplication since the test data is from a file
upvoted 1 times

savageboy 11 months, 1 week ago

Selected Answer: D

Test cases should added easily, that is keyword driven
upvoted 3 times

🗨️ 👤 **obtest123** 1 year ago

Probably, I would go with D. Keyword-driven scripting as far as initial cost is not under consideration and in Structured scripting test cases are not easy to add

upvoted 2 times

🗨️ 👤 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: B

B, C, and D would all eventually achieve the desired goals. However, the natural evolution for linear scripts (capture/replay) are structured scripts. Not to mention that the initial effort to create data-driven and keyword driven scripts would be much higher and costly.

upvoted 1 times

The Test Automation Manager has asked you to provide a solution for collecting metrics from the TAS that measures code coverage every time the automated regression test pack is run. The metrics must be trend based to ensure that the scope of the regression test pack continues to reflect enhancements made to the SUT - coverage must not drop and should ideally increase. The solution must be as automated as possible to avoid unnecessary manual overheads and errors.



Which of the following approaches would BEST meet these requirements?

- A. Test automation cannot measure code coverage for the SUT, only the code for the automation tools and scripts. The automated test cases would need to be run manually with a code coverage and reporting tool running in the background.
- B. The automated testware would record overall code coverage for each run and add the figure to a new row in a pre-formatted Excel spreadsheet. You would then present the spreadsheet to stakeholders so they could look for changes in coverage.
- C. The automated testware would record overall code coverage for each run, export the data to a pre-formatted Excel spreadsheet that automatically updates a trend analysis bar chart for you to distribute to stakeholders.
- D. The automated testware would record the pass/fail rate of each regression test case, export the data to a pre-formatted Excel spreadsheet that automatically updates a trend analysis success rate bar chart and emails it to stakeholders.

Suggested Answer: C

Community vote distribution

C (100%)

  **Od4fdcc** 9 months, 1 week ago

Selected Answer: C

The best approach for collecting metrics that measure code coverage is to automate the process as much as possible. Option C suggests an automated solution where the testware records the code coverage for each run and exports the data to an Excel spreadsheet. This spreadsheet is pre-formatted to update a trend analysis bar chart automatically. This method not only minimizes manual overheads and errors but also provides stakeholders with an easy-to-understand visual representation of the coverage trends over time. It ensures that the scope of the regression test pack is continuously reflecting the enhancements made to the SUT, with coverage metrics that should not drop and ideally increase, aligning with the requirements for trend-based metrics.

References = The ISTQB Test Automation Engineer syllabus and industry best practices recommend the use of automated solutions for tracking and analyzing test metrics. This includes the automation of metric collection and the use of trend analysis to monitor the effectiveness of the test suite over time¹²³.

upvoted 1 times

  **nemeremjcc** 10 months ago

Selected Answer: D

Stakeholders at management level focus on trends such as how many test cases were added since the last test run, changes in the pass-fail ratio etc.

upvoted 1 times

  **FrancisFerreira** 1 year, 1 month ago

Selected Answer: C

Alternative C seems correct.

upvoted 1 times

Which of the following is considered a disadvantage of test automation?

- A. Automated exploratory testing is difficult to implement
- B. Test automation can be a distraction from the objective of finding bugs
- C. Tests are more likely to have operator errors.
- D. Slower feedback on the quality of the system.

Suggested Answer: B

Community vote distribution

B (100%)

 **Od4fdcc** 9 months, 1 week ago

Selected Answer: B

Its B according to the Syllabus

Disadvantages of test automation include:

- ☒ Additional costs are involved
- ☒ Initial investment to setup TAS
- ☒ Requires additional technologies
- ☒ Team needs to have development and automation skills
- ☒ On-going TAS maintenance requirement
- ☒ Can distract from testing objectives, e.g., focusing on automating tests cases at the expense of executing tests
- ☒ Tests can become more complex
- ☒ Additional errors may be introduced by automation

upvoted 1 times

 **nemeremjcc** 10 months ago

Selected Answer: A

difficult to implement


upvoted 1 times

 **miguelliranzo** 10 months, 2 weeks ago

Selected Answer: A

A disadvantage of test automation is that automated exploratory testing, which is often used to find new and unexpected defects, is difficult to implement.

upvoted 1 times

 **anhtho** 1 year, 4 months ago

Selected Answer: B

In Syllabus obvious B

upvoted 2 times

 **savageboy** 1 year, 5 months ago

Selected Answer: B

In Syllabus obvious B

upvoted 2 times

 **FrancisFerreira** 1 year, 8 months ago

Selected Answer: B

Exploratory testing cannot be automated in any way. C and D are quite the contrary. B is the correct answer.

upvoted 4 times

New features have been added for the current release of a SUT.

Which action would NOT be appropriate for the TAE to perform when evaluating the impact on the TAS?

- A. Gather feedback from the Business Analysts to determine if the current TAS will meet the needs of the new features.
- B. Review existing keywords to see if they need to be modified.
- C. Run existing automated tests against the updated SUT to verify and record any changes to their correct operation.
- D. Evaluate compatibility with existing test tools and, where necessary, identify alternative solutions.

Suggested Answer: A

Community vote distribution

A (100%)

🗨️ **nemeremjcc** 10 months ago

Selected Answer: D

business analysts can be involved in the creation of automated test cases by following the Keyword Driven Test approach

upvoted 1 times

🗨️ **bberries** 12 months ago

Why is A the right answer here? It would be much more helpful to give a justification for your answer rather than just your answer

upvoted 1 times

🗨️ **savageboy** 2 years, 5 months ago

Selected Answer: A

A seems correct

upvoted 2 times

🗨️ **FrancisFerreira** 2 years, 8 months ago

Selected Answer: A

Alternative A seems correct.

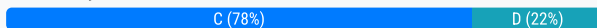
upvoted 2 times

You are implementing test automation for a project that has a business critical application. A test execution tool is being used to run automated regression tests. The results from the test execution tool are very important and need to be 100% accurate. You want to merge the test automation results with the test management system that also records the manual test results so that managers can make informed decisions about the progress quickly. Which layer of the gTAA will be used to ensure the proper reporting occurs and the interfaces to the test management system are handled?

- A. The reporting layer
- B. The logging layer
- C. The execution layer
- D. The adaptation layer

Suggested Answer: C

Community vote distribution



🗨️ **pabc08** 1 year, 3 months ago

Selected Answer: C

3.1.4 Test Execution Layer

The test execution layer consists of tool support for the following:

- ☑ Executing test cases automatically
- ☑ Logging the test case executions
- ☑ Reporting the test results

upvoted 2 times

🗨️ **anhto** 1 year, 10 months ago

Selected Answer: C

No reporting layer

upvoted 1 times

🗨️ **savageboy** 1 year, 11 months ago

Selected Answer: C

No reporting layer, and according to the syllabus reports generate in execution layer

upvoted 1 times

🗨️ **obtest123** 2 years ago

Selected Answer: C

Syllabus says Execution layer contains reporting results. Would go with C

upvoted 2 times

🗨️ **mustooo** 2 years ago

Selected Answer: C

In teh syllabus you have a picture of gTAA. You can see that the interface from the test execution layer can be done straight to test management layer. IMO test adaptation layer is to SUT.

upvoted 2 times

🗨️ **FrancisFerreira** 2 years, 1 month ago

Selected Answer: D

There is no gTAA layer called "reporting"... Reports are handled by the execution layer. Interfaces are handled by the adaptation layer. So, either C or D. Assuming that by "proper reporting" they refer to the consolidation of test results, I would go with D, coz the execution layer would gather the automation results, but the consolidation would ultimately be handled by the adaptation layer via the interface to the test management system.

upvoted 2 times

Which of the following describes how a test execution report is likely to be used?

- A. To understand which test step caused the failure in a test case
- B. To identify problematic areas of the SUT by keeping a history showing which test cases fail the most
- C. To measure coverage of the test basis by a test suite
- D. To record how a test case failure has been fixed

Suggested Answer: B

Community vote distribution

B (67%)

A (33%)

  **pabc08** 1 year, 1 month ago

Selected Answer: B

5.4 Test Automation Reporting

Option is to identify problematic parts of the SUT, is to keep a history of the reports, so that statistics about test cases or test suites with frequent regressions can be gathered.

upvoted 2 times

  **Leandro__** 1 year, 2 months ago

Selected Answer: A

A seems correct.

"Option is to identify problematic parts of the SUT, is to keep a history of the reports, so that statistics about test cases or test suites with frequent regressions can be gathered"

upvoted 1 times

  **FrancisFerreira** 2 years, 1 month ago

Selected Answer: B

Test reports dont usually contain historical data as they refer to a single test run (that is, each test case is only run once). But I guess you CAN implement a feature like that, so alternative B is as good as any. Actually B is slightly better, coz all the others are wrong.

upvoted 1 times

What is the PRIMARY advantage of using abstraction in the TAA?

- A. It makes it more flexible for future reuse and improvements
- B. It requires a higher skill level to implement
- C. It ensures that any scripting method will be supported
- D. It improves the performance of the TAS

Suggested Answer: A

Community vote distribution

A (100%)



 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: A

Alternative A seems correct.

upvoted 4 times

As the TAE, you are working with the organisation's Test Manager to decide which external metrics and which internal metrics should be gathered for the new TAS.

Which of the following represents the BEST internal metric that would help measure the quality of the TAS and the number of problems associated with the TAS?

- A. The average maintenance cost to keep an automated test in sync with the SUT
- B. The number of hours of manual test effort saved by implementing a TAS
- C. A measure of defect density within the TAS automation code
- D. A measure of how many automated tests pass and fail

Suggested Answer: C

Community vote distribution



 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: C

Alternative C seems correct.

upvoted 2 times

You are evaluating several test modelling tools and are wanting to automatically generate test cases within the tool where many different combinations of input data are created.

You are then wanting to export the test cases into a csv file which can then be read by a functional test execution tool using a data-driven or keyword-driven scripting method.

You have investigated several tools and there is only one tool that provides all the necessary features defined by your team with the exception of the export facility. It does not provide an export into either .xls or .csv formats.

What would be the BEST next step regarding the selection of this tool?

- A. Consider another tool that is more "fit for purpose" and has all the features required.
- B. Explore the possibility of creating your own export facility.
- C. Ask the vendor and use forums to see if a solution is available or going to be available in the future.
- D. Purchase this tool and generate the .csv file manually.

Suggested Answer: C

Community vote distribution

C (100%)

 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: C

Alternative C seems correct.

upvoted 3 times

You are testing a major enhancement to an air traffic control user interface. You have use of a sophisticated pre-production test environment, created specifically for large scale automated regression, performance and security testing. The window for regression testing is limited and must successfully conclude, with no major regressions remaining, before the non-functional testing starts.

You have been using the same version of the TAS for the last few releases, each time completing the automated regression test suite in a single overnight run. However, due to the latest enhancements for the SUT, you believe there is a risk that the test suite may no longer complete overnight and therefore delay performance and security testing.

Which option would be the BEST and MOST cost-efficient approach to mitigate this risk?

- A. Create a mirror of the pre-production test environment and split the regression test suite to run in parallel across the environments.
- B. Split the regression test suite into multiple parts, running in the environment across consecutive nights.
- C. Analyse the regression test suite and remove test coverage duplication and redundancy.
- D. Introduce better coding practices for the automation scripts, including coding guidelines, reviews and improved static analysis.

Suggested Answer: C

Community vote distribution

C (88%) 13%

 **FrancisFerreira** Highly Voted 2 years, 8 months ago

Selected Answer: C

A would solve the issue at hand, but so would C. Thing is, A is certainly the LEAST cost-effective, for creating a second top-notch pre-prod environment would NOT be cheap. For this reason I would go with C.

upvoted 7 times

 **Od4fdcc** Most Recent 9 months, 1 week ago

Selected Answer: C

In the context of testing a major enhancement to an air traffic control user interface with a limited window for regression testing, the best and most cost-efficient approach to mitigate the risk of the test suite not completing overnight is to analyze the regression test suite and remove test coverage duplication and redundancy (Option C). This approach streamlines the test suite by eliminating unnecessary tests, thereby reducing the overall execution time without compromising the test coverage's integrity. It ensures that each test case provides unique value and that the suite as a whole is optimized for efficiency. This method does not require additional resources or environments and leverages the existing test suite and infrastructure, making it a cost-efficient solution to address the potential delay in starting performance and security testing due to extended regression test durations.

upvoted 1 times


 **nemeremjcc** 10 months ago

Selected Answer: B

Time should be put into consideration here. Time is required to do further analysis of the test automation coverage, which will have to reveal duplication before attempting to remove it.

B is the answer

upvoted 1 times

 **eglitmar** 11 months, 2 weeks ago

B and C is correct, but if problem is also in time - then i would choose C.

upvoted 1 times

 **bberries** 12 months ago

Why wouldn't B be the most cost-efficient? e.g. If you had to investigate how many duplicated test cases there are, that could take up a significant amount of test resource depending on how many test cases/scenarios are in the solution. Some TASs have thousands of individual automated test scenarios in the one solution...

upvoted 1 times

 **issoChan** 1 year, 8 months ago

Selected Answer: A

Dublication is NOT mentioned in given explanation. So answer should include parallel test-> A

upvoted 1 times

If you are tracking the frequency that a test automation code reports a defect that is not really a defect, what metric are you gathering?

- A. Tool scripting metrics
- B. Automation code defect density
- C. Trend metrics
- D. The number of false-fail results

Suggested Answer: D

Community vote distribution



 **FrancisFerreira** 1 year, 1 month ago

Selected Answer: D

Alternative D seems correct.

upvoted 4 times

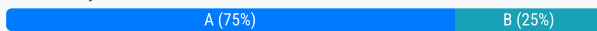
You are planning the pilot for an in-house developed Test Automation solution (TAS).

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. a and b
- B. b and d
- C. c and d
- D. c and e

Suggested Answer: A

Community vote distribution



bberries 12 months ago

Selected Answer: A

4.1.1

"It is important to DEFINE THE SCOPE of validation for a test automation pilot. A pilot project does not take a long time to conduct, but the outcome may have a significant impact on the direction that the project takes.

Based on the information gathered about the SUT and the REQUIREMENTS ON THE PROJECT, the following should be evaluated to set up guidelines to optimize the test automation efforts:

- Programming language(s) that will be used
- Suitable commercial off-the-shelf/open-source tools
- Test levels to cover
- Test cases selected
- Test case development approach"

i.e. every other consideration is SECOND to the consideration of choosing the appropriate pilot project

upvoted 2 times

pabc08 1 year, 7 months ago

Selected Answer: B

Plan the pilot

The pilot should be treated as a regular development project: make a plan, reserve budget and resources, report on the progress, define milestones, etc.

An extra attention point is to make sure that the people working on the TAS deployment (i.e., a champion) can spend enough effort on the deployment even when other projects demand the resources for their activities.

upvoted 1 times

Leandro_ 1 year, 8 months ago

Selected Answer: A



A seems correct to me.

upvoted 2 times

When the SUT provides insight into the behaviour of the system, providing the users the with the status of the various actions performed so that they can check that expected behaviour equals actual behaviour, what is this called?

- A. Portability.
- B. Maintainability.
- C. Observability.
- D. Controllability.

Suggested Answer: *C*

  **sctmp** 1 year, 1 month ago

C is correct



upvoted 2 times

You are a TAE working for a software house which provides quarterly releases of its software to its customers. There are many different versions of the SUT that need to be tested simultaneously by different tests teams.

Your TAS is complex and you need to ensure it remains consistent across the different SUT environments. What is the BEST and MOST efficient way to ensure each of the test teams use the same version of the TAS to test the different versions of the SUT?

- A. Due to the complexities involved and the high risks associated with these releases, it would be best to revert to manual testing.
- B. Produce comprehensive documentation of the TAS, installation and usage guidelines and provide training for each team member.
- C. Install the TAS in a central repository and have an automated installation and configuration of the TAS from this repository to each of the SUT environments.
- D. Develop a tool to track historical test results across the different SUT environments and look for trends.

Suggested Answer: C

  **sctmp** 1 year, 1 month ago

C is correct

upvoted 3 times

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.

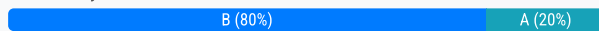
Initially these tests will be run manually to verify their correctness and ultimately you want to include them in the test execution tool so that they can run unattended.

Which layer of the gTAA will be used to support the specification of the test cases and preparation of the test data?

- A. The generation layer
- B. The definition layer
- C. The execution layer
- D. The adaptation layer

Suggested Answer: B

Community vote distribution



☒ **TheeOne** 10 months, 4 weeks ago

Selected Answer: B

Definition layer- define the tests which will then be generated automatically in the 'generation layer'

upvoted 1 times

☒ **eglitmar** 11 months, 2 weeks ago

Hmm... yes much text is about generation level, it could be definition level if these generated tests/ data want to implement in TAS...

upvoted 1 times

☒ **pabc08** 1 year, 7 months ago

Selected Answer: A

generate test cases automatically + generate test data automatically = Generation Layer

upvoted 1 times

☒ **sctmp** 2 years, 7 months ago

Selected Answer: B

B is correct

upvoted 4 times

You have been asked to implement test automation for a project that is not meeting its deadlines. After further analysis you discover that the manual testers are not able to keep up with the new feature testing because the regression testing is taking 75% of their time. As a result, the new features are being released with many defects and customers are complaining about the quality.

Given this information, what metric SHOULD you be tracking to show the value of test automation for this project?

- A. Percentage of code covered by the test automation.
- B. Equivalent Manual Test Effort for the automated tests.
- C. Number of defects found by test automation.
- D. Percentage of builds accepted/rejected by the automated tests.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Which of the following is an important success factor for any significant automation project?

- A. The TAA must be designed for testability.
- B. The TAA is self-documenting
- C. The SUT must be designed for testability
- D. The SUT is self-documenting

Suggested Answer: C

Community vote distribution


C (100%)

 **bberries** 12 months ago

Selected Answer: C

see syllabus 2.1.1 Describe the Configuration Needs of an Infrastructure that Enable Implementation of Test Automation

upvoted 1 times

 **sctmp** 2 years, 7 months ago

Selected Answer: C

C is correct

upvoted 1 times

Which of the following CORRECTLY describes how automation SHOULD be applied to confirmation testing?

- A. Confirmation tests are not good candidates for automation as they are not designed to run many times
- B. Confirmation tests should only be automated if they fail to pass on the first attempt
- C. Confirmation tests can be automated and incorporated into an automated regression suite to show whether defects that were previously fixed reoccur
- D. A confirmation test should only be automated after it has been run manually

Suggested Answer: C

Community vote distribution



 **obtest123** 1 year ago

Selected Answer: C

Agree with C

upvoted 1 times

You have inherited a TAS that is working well it uses keyword-driven scripting and was well architected. The automation architect who built the system has now moved on to another company. The TAS is working across several projects and has a multiple library of keywords, categorised by project. The individual project teams maintain these keyword scripts.

Based only on the given information, what is the MOST significant risk for the TAS?

- A. The keyword driven scripts may become out of date if not maintained
- B. The level of abstraction, coupled with the departure of the architect may make the system hard to maintain
- C. New projects may not work as well with the TAS as the current projects
- D. Because the keyword scripts are maintained by different teams, there is a likelihood that good coding standards are not followed

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Your company is new to test automation and as TAE, you have designed a TAS which successfully supports the SUT for the current project. There are other systems currently in operation which have been tested manually and more systems are planned over the coming years. Based on this success, your company requires test automation be rolled out to other current and future SUTs with consistency being a key objective. Which of the following is the BEST way to achieve that?

- A. Design a new TAS for each SUT, and manage each one through a dedicated automation support team.
- B. Install the current TAS into a central repository so that other tests on different SUTs use the same version of the TAS.
- C. Check for correct connectivity to internal and external systems to ensure that the TAS has been installed and configured correctly for each SUT.
- D. Develop a tool that keeps track of automation failures across the different SUTs and produces regular reports to stakeholders.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

When is the BEST time for automation to consider legal and/or standard requirements for a SUT?

- A. When implementing the SUT
- B. When designing a TAF
- C. When designing a TAA
- D. When developing a TAS

Suggested Answer: A

Community vote distribution

C (100%)

🗨️ **0d4fdcc** 9 months, 1 week ago

Selected Answer: C

Is correct. SUT standards and legal settings should be considered and any design requirements understood when designing the TAA
upvoted 1 times

🗨️ **anitanik** 1 year, 4 months ago

The correct answer here is C!

Ref from the syllabus: "In addition to the technical aspects of an SUT, the TAA design may need to respect legal and/or standards requirements so as to design the TAA in a compatible manner.", page 37 of 84

upvoted 3 times

🗨️ **sctmp** 2 years, 1 month ago

Selected Answer: C

the TAA design may

need to respect legal and/or standards requirements so as to design the TAA in a compatible manner.

upvoted 2 times

🗨️ **Gsha** 2 years, 1 month ago

I think C is correct. Legal standards should be considered when designing the TAA

upvoted 2 times

You are implementing a TAS for a system that has been live for over three years, using a hybrid waterfall and agile lifecycle. Live updates are made on a monthly basis.

There is no test team, with developers designing and executing unit and integration tests with some degree of automation and business analysts designing and executing manual tests at the system level. No formal test process exists, although the system has proved relatively stable for most of the time.

Unfortunately, the last two monthly releases were problematic with regression defects found in production. Your priority is the automation of functional regression tests at the system level, the budget for this has been approved by project stakeholders.

The Business Analysts have identified which test cases are most suitable for regression. You must use the organisation's long standing commercial automation tool which has passed a proof of concept in the platform for the system in question.

Which of the following suitability criteria needs the MOST attention for the TAS?

- A. Technical planning in support of ROI analysis
- B. Frequency of use.
- C. Compatibility and tool support
- D. Maturity of the test process

Suggested Answer: C

Community vote distribution

C (100%)

🗨️ 👤 **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: D

The Maturity of the test process (D) requires the most attention because there is no formal test process, which directly contributes to the recent production defects. The absence of a defined process makes any attempt at implementing and maintaining a new automation system very difficult, leading to potential failures and inefficiencies, making it the most critical factor to address first.

upvoted 1 times

🗨️ 👤 **Einram** 8 months ago

Selected Answer: C

C. Compatibility and tool support

upvoted 1 times

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?

- A. Move to a model-based approach to scripting where the models include the keywords.
- B. Do nothing, each team are working in isolation and they are all experiencing significant benefits in the way they are currently working.
- C. Provide each team with a set of guidelines and naming conventions for keywords.
- D. Create a central library of keywords and associated definitions for each team to use.

Suggested Answer: D

Community vote distribution

D (100%)

 **Einram** 8 months ago

Selected Answer: D

D. Create a central library of keywords and associated definitions for each team to use.

upvoted 1 times

You are working as a TAE for a company who are re-designing their website. The new website provides information for customers and there are two minor features being developed:

1. Request a newsletter
2. Ability to contact the organisation with a question or comment

The website must be "mobile friendly" and available on all major web browsers.

You have been tasked to provide an automated solution for web browsers only and to concentrate on the two minor features.

What would be a KEY challenge with automation in this context?

- A. A low level of intrusion is likely from use of existing UI elements, but depending on the solution this might be more complex than a higher level of intrusion.
- B. Because there is a high level of intrusion there may be many false alarms.
- C. Automation might not be possible on the mobile devices.
- D. The benefits of automation might not be achieved for many years due to the complexities of the SUT and automation solution.

Suggested Answer: D

Community vote distribution

A (100%)

🗨️ 👤 **Svarog** 1 year, 3 months ago

Selected Answer: A

A. A low level of intrusion is likely from use of existing UI elements, but depending on the solution this might be more complex than a higher level of intrusion.

Explanation:

In this scenario, the focus is on automating the testing of two minor features on a redesigned website, ensuring mobile-friendliness and compatibility across major web browsers. Option A addresses the challenge associated with the level of intrusion required for automation. While a low level of intrusion is expected when utilizing existing UI elements for automation, it may still pose complexity depending on the specific automation solution chosen. Implementing automation that interacts with UI elements may require careful scripting and handling to accurately simulate user interactions and validate the functionality of the features.

upvoted 2 times

You are the TAE for an Agile project which has six sprints for the current release. Sprint five is underway and the automated regression suite is due to start later today.

You have re-examined the results from the automated regression runs for the past four sprints. You notice that two test cases both reported a pass for sprints 1 and 4 but a fail for sprints 2 and 3. The failures have gone undetected and are therefore unexplained. Both test cases are closely coupled with other tests in the suite.

What course of action SHOULD you take?

- A. Run the regression suite as planned and see if the tests fail again if they do, determine the cause of failure.
- B. Remove the test cases from the regression suite and refer them to the test designer for manual testing for future sprints.
- C. Quarantine the test cases, run the regression suite without them, and perform root cause analysis on the test cases in parallel.
- D. Run the test cases manually now. If they still pass, keep them in the regression suite, if they fail perform root cause analysis.

Suggested Answer: C

Community vote distribution

D (100%)

- 🗨️ 👤 **TreyQA** 1 year ago
anything to back that up Leandro?
upvoted 2 times
- 🗨️ 👤 **Leandro__** 1 year, 2 months ago
Selected Answer: D
D seems correct for me
upvoted 1 times
- 🗨️ 👤 **Leandro__** 1 year, 2 months ago
D seems correct for me
upvoted 1 times

In order to achieve re-use of a TAS, where SHOULD the design for reuse occur?

- A. At the code level
- B. At the framework level.
- C. At the TAS level
- D. At the TAA level

Suggested Answer: C

Community vote distribution

D (100%)

🗨️ **issoChan** 1 year, 2 months ago

In section 3.3.4 this is written:

"While reuse aspects are already settled when the TAA is defined, the TAS can help increase the ability for reuse."

So answer should be D

upvoted 1 times

🗨️ **pabc08** 1 year, 3 months ago

Selected Answer: D

"It is important to note that while design for reuse is mainly a matter for the TAA...", page 43

upvoted 1 times

🗨️ **anitanik** 1 year, 4 months ago

Correct answer is D!

The question is asking where the design for reuse occur - in the TAA!

Ref from the syllabus: "While reuse aspects are already settled when the TAA is defined", page 42 of 84.

upvoted 3 times

🗨️ **sctmp** 2 years, 1 month ago

Selected Answer: D

It is important to note that while design for reuse is mainly a matter for the TAA, the maintenance and improvements for reuse are a concern throughout the TAS lifecycle.

upvoted 1 times

🗨️ **sctmp** 2 years, 1 month ago

answer C *

upvoted 1 times

You have executed an automated test suite for a product that was released into production. Although all the tests passed, there was a major failure in production in an area that was covered well by your automated tests.

You have run the automated tests again and one of the tests is now failing and this is directly related to the production defect that was raised. You decide to run the automated test suite again on the same version of the SUT and the test now passes.

What SHOULD you do now to verify the validity of the automated tests?

- A. Remove the intermittently failing test from the test suite and investigate the reason why the test sometimes passes and sometimes fails.
- B. Check that the production defect that was reported was an actual defect
- C. Run the automated test suite again and if the test now passes - do nothing
- D. Remove the intermittently failing test from the test suite and re-run the automated test suite again

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!


What are the four horizontal layers of the gTAA?

- A. Test adaptation, test execution, test design, test definition
- B. Test generation, test execution, test definition, test APIs
- C. Test generation, test definition, test execution, test adaptation
- D. Test definition, test execution, test reporting, test adaptation

Suggested Answer: C

Community vote distribution

C (100%)

 **sctmp** 1 year, 1 month ago

Selected Answer: C

C is correct

upvoted 2 times

Your organisation has successfully implemented a Test Automaton Solution (TAS) for a new project which has since been delivered into production via a number of sprints. A series of maintenance releases are now planned.

Some improvements were made to the Test Automation Architecture (TAA) as a result of feedback from the early sprints. The TAA improvements affected the TA, and the TAS was changed for the final sprint.

The new version of the TAS was generally well received but some performance and usability issues were encountered with the TAS which have yet to be addressed.

The test automation engineers supporting the maintenance releases must decide whether to use the enhanced TAS or the version that was used successfully for previous sprints.

What is the BEST action to take next?

- A. Perform an analysis of risks versus benefits for the enhanced TAS and then decide which version to use.
- B. Use the previous version because this was proven to work. It will be too risky to use the new version, with unresolved issues, for a live system.
- C. Use the new version because, despite some issues, it works, and the live system should not be tested using a different TAS.
- D. Use the new version of the TAS for the first maintenance release on a trial basis. If issues are encountered, switch to the previous version for later releases until the issues are resolved.

Suggested Answer: D

Community vote distribution

D (50%)

A (50%)

🗨️ **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: A

The BEST action is A. Perform an analysis of risks versus benefits for the enhanced TAS and then decide which version to use. This approach allows for a structured evaluation of the new TAS's potential value against its known issues, enabling an informed decision based on project needs and risk tolerance, rather than blindly choosing the previous version (B), the new one (C), or a risky trial (D).

upvoted 1 times

🗨️ **cloudobject** 9 months, 3 weeks ago

Selected Answer: D

D is the answer, running it on the enhanced TAS will and if it fails, it is switched will provide sufficient result for root cause analysis for a fix, while it runs on the old version.

upvoted 1 times

🗨️ **issoChan** 1 year, 2 months ago

Selected Answer: D

See my explanation below

upvoted 2 times

🗨️ **issoChan** 1 year, 2 months ago

In A, first action to be done is mentioned as in given text->> Perform an analysis of risks versus benefits for the enhanced TAS and then decide which version to use.

And asked what is NEXT?

So answer is then D.

upvoted 1 times

🗨️ **Svarog** 1 year, 3 months ago

Selected Answer: A

Correct answer is A

See Syllabus:

When maintenance deployment occurs, the following steps are needed:

- ☑️ Make an assessment of the changes in the new version of the TAS compared to the old one
- ☑️ Test the TAS for both new functionality and regressions
- ☑️ Check if the test suite needs to be adapted to the new version of the TAS

upvoted 2 times

🗨️ 👤 **anitanik** 1 year, 4 months ago

Correct answer is A!

Ref from the syllabus: "The updated TAS has additional defects or performance issues: perform an analysis of risks vs. benefits.", page 48 of 84

upvoted 1 times

🗨️ 👤 **Gsha** 2 years, 1 month ago

Why not A ?

upvoted 1 times

What represents good practice when automating a manual regression test suite?

- A. Test data shared between tests should, where feasible, be stored and accessed from a single source to avoid duplication or introduction of error.
- B. All existing manual tests should be decomposed into several smaller automated tests to reduce functional overlap.
- C. Remove inter-dependencies between tests to reduce automation failures and costly error analysis.
- D. Once a manual test has been automated, execute it immediately to identify whether it operates correctly.

Suggested Answer: A

Community vote distribution

A (100%)

🗨️ 👤 **Wokawokaeee** 1 year, 4 months ago

Selected Answer: A

should be A, i don't find nothing for when to perform test on syllabus.

upvoted 2 times

🗨️ 👤 **zaazoozeba28** 1 year, 4 months ago

However, in an automated test, data which is shared should—where possible and feasible—be stored and accessed from a single source to avoid duplication, or introduction of errors.

upvoted 2 times

🗨️ 👤 **sctmp** 2 years, 1 month ago

Selected Answer: A

. However, in an automated test, data which is shared should—where possible and feasible—be stored and accessed from a single source to avoid duplication, or introduction of errors.

upvoted 1 times

You have investigated a new tool which enables the modelling of the SUT and can then generate test cases either manually or automatically. You have convinced your managers that the best way forward is to conduct a pilot project for this tool. You need to select a project to use for the pilot. You have the choice of the following projects:

Project A: A two-year project that is critical to the business and is currently in the requirement phase. This project is for a new e-commerce web site and is mostly being developed "in-house" although the payment system is being developed and delivered by a 3rd party provider.

Project B: A safety critical application for software to drive and park cars.

Project C: An upgrade to an important HR timesheet tracking application that will be available on a desktop and mobile application. This is a 1-month project developed in-house.

Project D The payment system from project A.

Which project would be BEST for the pilot?

- A. Project A because it is a large project and has high visibility and is in the requirement phase.
- B. Project B because it is a safety critical system and has high visibility.
- C. Project C because it is a short, low priority project but is important
- D. Project D because it is a small part of a larger project and will help show the tool's capabilities.

Suggested Answer: D

 **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: C

Project C, the 1-month upgrade to the HR timesheet tracking application, is the best choice for the pilot project. This shorter project is less risky, allows for rapid feedback on the new modeling and test generation tool, and focuses on an in-house, well-understood system, making it ideal for quickly evaluating the tool's effectiveness before applying it to larger, more complex, or external projects.

upvoted 1 times

 **anitanik** 1 year, 4 months ago

The correct answer is C!

A and B are wrong - as these are all critical projects and as noted in the syllabus - "Do not select a critical project. When the deployment of the TAS causes delay, this should not have major impact on critical projects.", page 45 of 84.

In answer A is written that the payment system, suggested in answer D, is developed and delivered by a 3rd party provider. So, selecting this option for a pilot project will not have any added value.

Answer C suggests short, low priority but important project. There's no saying it's trivial!

upvoted 3 times

 **rajlu717** 1 year, 5 months ago

C looks to be correct answer.

upvoted 1 times

Your project is transitioning from manual to automated testing. You have decided to implement a pilot project so that lessons learned can inform future time estimates and schedules.

Which two of the following represent the types of test cases that are MOST suited to a test automation pilot project? a) High added value test cases that require little effort to automate. b) Test that are run infrequently as these will be simpler to automate c) Reliability test cases that can show added value soon d) Technically challenging test cases to provide the best validation of manual test conversion e) Tests that are least important to the business as these are safer to trial

- A. a and b
- B. a and c
- C. b and d
- D. c and e

Correct Answer: B

 **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: A

The best answer is A. a and b.

Explanation:

High added value test cases that require little effort to automate:

This is ideal for a pilot project because it allows you to quickly see the benefits of automation with minimal investment.

Tests that are run infrequently:

These are also good candidates as they can be automated and then run less frequently, still providing valuable coverage without significantly impacting the manual testing workload

upvoted 1 times

You have been asked to develop test automation for a legacy system that is going to go through a series of infrastructure migrations. The scripts will be used to verify basic functionality during these infrastructure changes. Your Test Analysts have some programming skills and need a solution that is simple and fast.

Maintainability of the scripts is not a consideration because no changes to the software are anticipated.

Which of the following is the BEST scripting approach in this situation?

- A. Structured scripting
- B. Capture-replay scripting
- C. Model-Based scripting
- D. Linear scripting


Suggested Answer: B

 **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: D

The best scripting approach is D. Linear scripting because it is simple, fast to implement, and suitable for verifying basic functionality in a stable, unchanging legacy system where script maintainability is not a concern. Capture-replay is an option but generally produces brittle tests. Model-based is too complex for this use case, and structured scripting would be overkill given the system's immutability.

upvoted 1 times

 **cloudobject** 9 months, 3 weeks ago

Selected Answer: B

For a QA with some knowledge of programming, Capture-replay scripting is the right choice, it requires little to know programming experience compared to Linear scripting which the QA lacks the adequate knowledge to write a script end to end.

upvoted 1 times

 **bberries** 12 months ago

The question includes the following detail "Your Test Analysts have some programming skills"

...in light of this clue, why wouldn't Linear scripting be the right choice, as this scripting technique is "Easy to set up and to start writing test scripts" (according to 3.1.4) ?

upvoted 1 times

You are working on a web-based application called Book Vault that allows people to upload books and order books. This application must be available on all major browsers.

You have been testing the application manually and management have asked you to consider automating some of the tests.

You have investigated a number of commercial and free tools which can automate tests at a web browser level and one tool in particular meets your requirements and you have implemented a trial version.

You have basic programming skills and the main goal is to automate a few functional tests to see if the tool is compatible with the application and can recognise the objects and controls.

Which scripting technique would be MOST suitable in this scenario in order to meet the objectives?

- A. Structured scripting
- B. Capture-replay scripting
- C. Data-driven scripting
- D. Model-based scripting

Suggested Answer: B

  **bberries** 12 months ago

Why wouldn't the answer here be Structured Scripting?

The question mentions nothing at all about needing a solution delivered quickly or easily... but the question does specifically say that "You have basic programming skills"...

upvoted 1 times

Which of the following is not an advantage of test automation?

- A. The ability to perform tests which would be difficult or impossible to execute manually
- B. The ability to run more tests in less time and therefore to make it possible to run them more often
- C. The ability to find more defects with the same tests compared to the manual execution
- D. The ability to enable a better use of skilled testers by freeing them from repetitive and boring tasks

Correct Answer: C

  **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: D

The option that is not an advantage of test automation is D. The ability to enable a better use of skilled testers by freeing them from repetitive and boring tasks; while automation can certainly reduce repetitive tasks, it often requires skilled testers to design, develop, and maintain the automation framework, making them more involved in higher-level testing activities, not necessarily freeing them from all repetitive work.

upvoted 1 times

Assume that you are working on a TAS for standalone desktop applications. The automated tests are developed based on a given automation framework that allows interacting with the application using the GUI elements by means of an object-oriented API. The TAS needs also a tool that allows identifying the properties of these elements. In this tool the root element represents the current desktop and the child elements represent application windows. Each of these child elements contains GUI elements (e.g., menus, buttons, radio buttons, text boxes, toolbars etc.) with their properties. Unfortunately, the elements of two third party components are not identifiable by this tool and thus they cannot be used with the specified framework.

Based only on the given information, which of the following is the first step that you should consider to address such issue in this scenario?

- A. Verify the testability support within the current automation solution with the providers of the third party controls
- B. Verify whether the identifiability issue associated to the third party controls depends on the web browser
- C. Adopt an approach that uses the coordinates of the GUI elements for interacting with the application
- D. Verify whether naming standards for variables and files have been defined for the current automation solution

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

Which of the following best refers to the characteristic of observability associated with the (design for) testability of the SUT?

- A. The ability of the SUT to perform its intended function for a specified period of time
- B. The ability to exercise the SUT by entering inputs, triggering events and invoking methods
- C. The ability of the SUT to prevent unauthorized access to its internal components or data
- D. The ability to identify states, outputs, intermediate results and error messages in the SUT

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

You are currently designing the TAA of a TAS. In particular, 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 system that is described by a finite-state machine and exposes its functionality via an API. Assume also that the behavior of the SUT depends on possibly unreliable hardware or communication links.

Based only on the given information, which of the following aspects would you expect to be most important when designing the TAA in this scenario?

- A. Looking for tools that allow directly denoting exceptions and actions depending on system events
- B. Adopting a test generation strategy based on classification trees coverage for the test generation layer
- C. Looking for tools that allow performing setup and teardown of the test suites and the SUT
- D. Adopting a test generation strategy based on use case/exception case coverage for the test generation layer

Suggested Answer: A

  **Leandro_** 1 year, 2 months ago

Is c alternative correct?

upvoted 1 times

Consider the following layers of the gTAA structure:

I) Test generation layer -

II) Test definition layer -

III) Test execution layer -

IV) Test adaptation layer -

Consider also the following capabilities associated with these layers:

- 1) Acquire all the necessary resources before to run each test and release them all after the run, in order to avoid interdependencies between tests
- 2) Allow the automated test scripts on an abstract level to interact with components, configurations, and interfaces of the SUT
- 3) Define test directives that allow configuring the algorithms used to produce automatically the test cases from a given model of the SUT
- 4) Allow the specification of abstract and concrete test cases for example by means of templates and/or guidelines

Which of the following best matches each layer with the appropriate capability?

- A. I-3, II-4, III-1, IV-2
- B. I-4, II-3, III-1, IV-2
- C. I-4, II-3, III-2, IV-1
- D. I-3, II-4, III-2, IV-1

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

Assume that, as part of the design choices made for a TAA, 3-switch coverage is the technique that has been selected for determining which test cases have to be produced for automated model-based testing.

Which of the following layers of the gTAA does this design choice refer to?

- A. Test generation layer
- B. Test definition layer
- C. Test execution layer
- D. Test adaptation layer

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

Which of the following statements, related to aspects of the SUT to be considered when designing a TAA, is mostly true?

- A. All the interactions between SUT and TAS should be logged with the highest level of detail
- B. All the internal test interfaces of the SUT should be removed prior to the product release
- C. All the interfaces of the SUT affected by the tests should be controllable by the TAA
- D. All the external test interfaces of the SUT should be removed prior to the product release

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Assume that you are the TAE responsible for the correct functioning of a TAS deployed in a test environment that consists of a few machines running the same version of the operating system. The TAS has been working and stable since its deployment, and it has been used to run an automated test suite consisting of many similar automated tests. The infrastructure team is planning to update the operating system on these machines by installing a new service pack for security reasons. Since the vendor of the operating system assures full backward compatibility, the infrastructure team assures that there will be no impacts on the functioning of the TAS.

Based only on the given information, which of the following approaches would you adopt for confirming the functioning of the TAS with respect to the specified update of the test environment?

- A. Verify the behavior of the automated tests starting from a reduced set of them and then proceed incrementally until confirming the correct functioning of the whole automated test suite
- B. Make sure that the infrastructure team will install the service pack also on the machines where the SUT is running and then run the whole automated test suite to check its behavior
- C. Verify the behavior of the whole automated test suite by running all the automated tests since you should not expect any failures based on the infrastructure team assurance that there will be not impacts on the TAS
- D. Do not run any test because you can immediately confirm the correct functioning of the automated test suite since the infrastructure team assures that there will be not impacts on the TAS

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

The GUI of a CRM (Customer Relationship Management) application has been delivered through Internet Explorer with proprietary Active X and Java controls. This implementation allows enabling rich client capabilities, but specific commercial test automation tools are necessary for the purpose of automating test cases at GUI level. You have basic programming skills and your main goal is to quickly produce at a low cost some automated scripts, by automating a small number of functional test cases, for demonstrating only whether a small set of those commercial tools are able to properly recognize the actions taken by a tester when interacting with the GUI of the CRM application.


Based only on the given information, which of the following scripting techniques would you expect to be most suitable in this scenario?

- A. Data-driven scripting
- B. Keyword-driven scripting
- C. Linear scripting
- D. Structured scripting

Suggested Answer: C

Community vote distribution



 **pabc08** 1 year, 3 months ago

Selected Answer: C

Considering the goals of quickly producing low-cost scripts with basic programming skills and limited testing scope, the most suitable scripting technique for this scenario is C. Linear scripting

upvoted 3 times

You are using a gTAA to create a TAS for a project. The TAS is aimed specifically at automating given sets of manual test cases for standalone desktop applications. All the interactions between TAS and SUT will involve only the GUI of such applications.

Based only on the given information, which of the following components of the gTAA would you include in the TAS?

- A. The protocol component of the test adaptation layer
- B. The test model component of the test generation layer
- C. The API component of the test adaptation layer
- D. The test reporting component of the test execution layer

Suggested Answer: B

Community vote distribution

D (100%)

🗨️ 👤 **HiteshaPatel** 1 month, 2 weeks ago

Selected Answer: C

C. The API component of the test adaptation layer in the TAS, because the TAS needs to interact with the SUT's GUI to automate the test cases, and a GUI API would facilitate this interaction within the framework of a test automation framework (TAF) built on a gTAA's principles.
upvoted 1 times

🗨️ 👤 **0d4fdcc** 9 months, 1 week ago

Selected Answer: A

In the context of using a gTAA for creating a TAS, particularly for automating a suite of existing manual test cases for standalone desktop applications with interfaces from the CUI of the application, the focus should be on the Test Adaption layer. This layer is crucial as it deals with adapting the test cases to the specific interfaces of the System Under Test (SUT). It involves creating the necessary mappings and interactions with the CUI, ensuring that the automated tests can effectively communicate with the application as a user would through the CUI.
upvoted 1 times

🗨️ 👤 **pabc08** 1 year, 3 months ago

Selected Answer: D

Standalone desktop applications don't typically involve network protocols, APIs, or complex models, eliminating options A, B, and C.
upvoted 2 times

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

Based only on the given information, which of the following components of the gTAA would you exclude from the TAS in this scenario?

- A. The test reporting component of the test execution layer
- B. The test model component of the test generation layer
- C. The test execution (test engine) of the test execution layer
- D. The CLI component of the test adaptation layer

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!



Which of the following statements about the reuse of TAS artifacts is true?

- A. Reusable TAS artifacts can include components (or part of them) associated with different layers of the TAA
- B. To enable reuse of TAS artifacts a good design for reuse built into the TAA is needed and no further actions are needed during the TAS lifecycle
- C. Continuous maintenance and improvements for reusing TAS artifacts are mostly addressed during the design of the TAA
- D. Reusable TAS artifacts associated with the definition layer of the TAA include the adaptors to the SUT components and/or interfaces

Suggested Answer: C


Community vote distribution

A (100%)

  **pabc08** 1 year, 2 months ago

Selected Answer: A

3.3.4 Building Reuse into the TAS

Reuse of a TAS refers to the reuse of TAS artifacts (from any level of its architecture) across product lines, product frameworks, product domains and/or project families. Requirements for reuse result from the relevance of TAS artifacts for the other product variants, products and/or projects. Reusable TAS artifacts can include:  (Parts of) test models of test goals, test scenarios, test components or test data.

upvoted 2 times

A regression test suite consists of 500 test cases which are all executed manually. The business case for a pilot project is the adoption of test automation based on a commercial tool for reducing the execution time by a factor of 90% for 100% of the tests of such regression test suite. The pilot project lasted one month as planned and you are currently evaluating its results. At the end of the pilot project 40% of the regression tests have been automated and their execution time has been reduced by a factor of 60%.

Based on the given information, which of the following statements would you expect most likely to be true in this scenario?

- A. The duration of the pilot project has been too short. The pilot project should last until the success factors defined in the business case are achieved
- B. The target defined for the business case is overly accurate. The target for a pilot project should not be measurable
- C. The project selected for the pilot is too critical. The project selected for a pilot project should be not too critical or too trivial
- D. The target defined for the business case seems difficult to hit. The target defined for the pilot project should be realistic

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Consider a pilot project for a test automation tool. You have identified a suitable project, planned and conducted the pilot. The pilot has been successful and the tool is being deployed within your organization, following an approach that prescribes to increase tool use by one project at a time. During this rollout some test processes could be slightly changed to gain additional benefits from using the tool. As a result of the pilot project a reduced set of manual tests have been automated for the first time. You are currently monitoring the test automation efficiency and such monitoring reveals that the automation regime for such tests is not yet mature.

Based only on the given information, which of the following statements would you expect most likely to be true?

- A. The approach used for deploying the tool within the organization seems compatible with the guidelines that refer to the success factors for deployment
- B. The test execution time will benefit more than other aspects (e.g., failure analysis) from the evolution to a more mature automation regime
- C. The target defined for the pilot project was inappropriate, because the automation regime for the automated tests at the end of the pilot is not yet mature
- D. During the rollout, test processes should not be modified to gain additional benefits from using the tool. Tool use should be adapted to fit the processes

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

Consider a TAS that is going to be deployed into production for the first time. The TAS requires shared resources and has been developed specifically to run in a single and specific target environment separated from the SUT environment. The infrastructure in which the TAS will run has been created as well as the procedures for maintaining the TAS and its infrastructure, and the suite that the TAS will execute. Very unlikely the TAS will be required to work in new unknown environments over the next three years (its expected operational lifetime) while there is a high-risk that after the TAS will be deployed in the target environment, a number of existing applications could no longer work because of conflicts with the existing shared resources. The TAS will be subjected to many corrective, adaptive and perfective maintenance activities during its lifetime.

Based only on the given information, which of the following activities would you expect to be most effective at mitigating the risk associated with the first deploy into production?

- A. Testing the TAS for compatibility issues associated with the target environment
- B. Testing the TAS for its ability to be transferred into different target environments
- C. Testing the TAS for regressions due to optimizations that fix non-functional issues
- D. Testing the TAS for its ability to run in the same environment of the SUT

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!

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 the GUI. A keyword-driven framework has been adopted for automating the regression tests. Such tests are based on an identification at low-level of the web page components (e.g., class indexes, tab sequence indexes, and coordinates). In the next planned releases, the SUT will be subjected to a significant corrective maintenance (bug-fixes) and evolution (new features). Maintenance costs needed to update the test scripts should be as low as possible and high reusability of test scripts is also expected.

Based only on the given information, which of the following statements would you expect most likely to be true in this scenario?

- A. The keyword-driven framework is not suitable. It would be better to adopt a structured-scripting approach
- B. False positive errors are likely to occur when running the automated tests on the new releases without modifying the tests
- C. The total execution time of the automated regression test suite will increase progressively for the next planned releases
- D. The keyword-driven framework introduces a level of abstraction that is too high and makes difficult to understand what really happens

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Which of the following statements does not describe a good practice for maintaining the TAS?

- A. TAS must run in the development environment because development and programming knowledge are required for its maintainability
- B. The TAS must be under configuration management, along with the test suite, the testware artifacts and the environment in which it runs
- C. The TAS must separate the test scripts from the environment in which it runs and from the associated harnesses and artifacts
- D. The TAS must consist of components that can be easily replaced without affecting the overall behavior of the TAS itself

Correct Answer: A

Currently there are no comments in this discussion, be the first to comment!