Test Automation Planning

Get started with automation

If you’re new to automation tools or just getting started with test automation, these points can help you get started.

1. Understand the tool and create a plan

Familiarize yourself with Zoho QEngine. Know where to create a test case and how to automate it. If you understand how the product works, you’ll be aware of what you can achieve and the information you need for it. Learn about the application or the website or whatever you intend to test. This will give you an idea of what really needs to be tested and if it can be automated. A simple plan like this is a good way to start.

2. Identify areas that need testing and see if they can be automated

Testers are aware of test cases, which are basically a sequence of actions to check if an app or a component of the app is working as expected. If you were already performing such test cases manually, you can start with this list to see which of them can be automated.

Such tests can be classified into:

- **Repetitive tests**: Tests that have to be performed a number of times and can easily be scripted. For example, filling up a number of fields for various forms. In this case, you'd first want to check if the form loads properly. Check if the mandatory and optional fields work as expected. There could be multiple other scenarios that have to be repeated for each form.

- **Frequent tests:** Tests that require frequent runs, especially when data is involved. A simple example could be testing attachments of various types, of different sizes, and from multiple locations.

- **Time-consuming tests:** Tests that take up a lot of time are a good pick for automation. Take the simple example of signing up for an application, which has to be performed in different browsers, platforms, and environments.

- **High risk or crucial tests:** Tests that directly impact the business. This doesn't mean only major committed updates that have to be released on time, but also activities that follow a set of rules or guidelines; for example, privacy. Tests involving real-time data are better off automated than letting an individual have access to the data.

- **Interface tests:** UI tests, such as clicking or double-clicking buttons, or even waiting for a particular amount of time before performing the next action, can be easily automated using simple commands like “click” and “wait.”

If you don’t have any such identifiable tests, you can decide to test every part of your system systematically or try categorizing your system following the points mentioned above. You can also consider processes that are more prone to risk or failures, and likely to contain bugs. Irrespective of the approach you take, the most important point is to get started no matter how simple or small the tests are. Once you have hands-on experience, you’ll eventually figure out how to scale to larger testing processes. An additional point to remember is that not all tests can or should be automated. Ruling out such tests could also be a starting point until you decide on if and how the remaining tests can be automated.

Once you have an idea of the product, and you have a starting point in terms of what needs to be tested, begin with creating test cases.

3. Create test cases

Test cases are nothing but the actions that the test will perform. This is where you’ll provide the instructions for what needs to be tested. Divide tests into individual parts. Each part must focus only on one activity; for example, in a login process, loading the website is one activity. It's easier to debug a single activity compared with debugging a process containing multiple activities. Not just that, the test can be reused, for instance, for other platforms.

4. Group test cases into a suite and plan or schedule the automation

After creating individual tests, group them into a suite so that you can run them together or in a sequence. This lets you stitch together activities that are different but belong to a single process, and therefore need to be run in a chain. Set up various settings such as the browser or the platform on which they need to run, and schedule the automation.

5. Create elements to save frequently used actions

Some actions seem repetitive when you test web or mobile applications; say, when we try checking out a cart after a purchase or submitting a form online. To avoid writing the same lines of script for these actions, we can save their locators as elements to reuse them in future.

6. Record test cases

Instead of manually scripting test cases, record them for ease. Recording your test cases not only saves time; missing any of the actions performed on each applications or website can be avoided. Every click and scroll has been saved.

Additional points to consider

- Your test data refers to the information used during the testing phase to evaluate the functionality and performance of an application or system. With Zoho QEngine, your test data remains private—we do not monitor or modify it during testing. However, to enhance security and privacy, it’s always best to avoid using real or sensitive information.
- Decide if you want to run tests locally or on the cloud, which is the default. Local testing allows you to run the tests on your machine. This gives you control of the environment in which the tests run, giving you some form of assurance. However, this approach misses a minor advantage that comes from testing on the cloud, which, in a way, also performs some indirect load and performance testing because it connects to various online endpoints and traffic. You can choose any option depending on your requirements.
- Tests such as exploratory testing, where one simply explores an app, cannot be automated because it requires an actual human approach. Usability and accessibility testing, by definition, need human testers, and, as a result, aren’t recommended for automation.
- Testers who prefer using the local network for testing before moving their websites or applications to actual internet can integrate their network with Zoho QEngine's cloud.

Conclusion

Overall, while automation may seem a bit overwhelming at first, it actually makes the testing process easier and more efficient. Zoho QEngine is a simple software to test your software. If you can manage the process of developing an entire software by yourself, you shouldn’t find it difficult to automate the process of testing it. Begin with simply exploring, or create a few simple test cases. With the features of Zoho QEngine at your disposal, you can scale your test automation to a large extent within no time.