Automated Testing

Overview

Lecturer: Nguyen Thanh Quan (MSc)
Email: tg_nguyenthanhquan_cntt@tdtu.edu.vn

Content

- Introduction to Automation Testing
- Need for Automation Testing
- When to Automate?
- When is Automation an added benefit to Manual Testing?
- Which Tests to Automate?
- Types of Automation Testing
- Tools Available for Automation Testing
- When Test Automation is not the best solution?
- Advantages of Automation Testing
- Limitations of Automation Testing
- Challenge of Automation Testing

Introduction (1)

- What is Automation?
  - Automation Testing is **use of tools or software** to perform software testing.
  - **Developing** and **executing** tests that can run and compare the actual to **expected results**.
  - The automation software can also enter **test data** into the system under test, **compare expected and actual results** and generate detailed test **reports**.

Introduction (2)

- Why to Automate?
  - Automation **improves accuracy**, **increases coverage**.
  - To make test execution **faster**, **accurate** and improves **quality.**

- How To Automate?
  - Automation Test Tools

Need for Automation Testing

- **Speed**: Automation scripts are fast when compared to manual testers efforts.
- **Reliable**: Tests perform precisely the same operations each time they are run, thereby eliminating human error.
- **Repeatable**: tests can be repeated n number of times for execution of the same operation
- **Coverage**: Automated tests increase coverage
- **Reusable**: We can reuse tests on different versions of an application, even if the user interface changes

When to Automate Tests?

- **Regression Testing**: When the software application is fairly stable and only regression tests need to be executed
- **Smoke Testing**: For getting a quick high-level assessment on the quality of a build and making go/no-go decision on further testing
- **Static & Repetitive Tests**: For automating testing tasks that are repetitive and relatively unchanging from one test cycle to the next
- **Data Driven Testing**: For testing application functions where the same functions needs to be validated with lots of different inputs and large data sets (i.e. login, search)
- **Load & Performance Testing**: No viable manual alternative exists

When Automation has benefits?

- **Long Run projects:** Automated tests can run fast and frequently, which is cost-effective for software products with a long maintenance life
- **Loads of Regression Testing**: Reuse tests, which is good for regressions on constantly changing code
- **When manual Tests are time consuming and complex**: Time constraints often make it impossible to manually test every feature thoroughly before and software application or web application is to be releases. This leaves us wondering if serious defects have been detected or not
- **Application is stable and doesn’t have frequent GUI changes**

Which Tests to Automate?

- **Business Critical test cases:** Business Scenarios and all critical flows which needs to be perfectly tested and function.
- **Smoke Tests**: Smoke Tests are good candidates for Automation
- **Test Cases that are very difficult to perform manually**: Task that involve complex calculation, tedious steps are good candidates for automation
- **Test Cases or Modules which are stable enough**: If the application or module is not stable enough, it is not worth automation

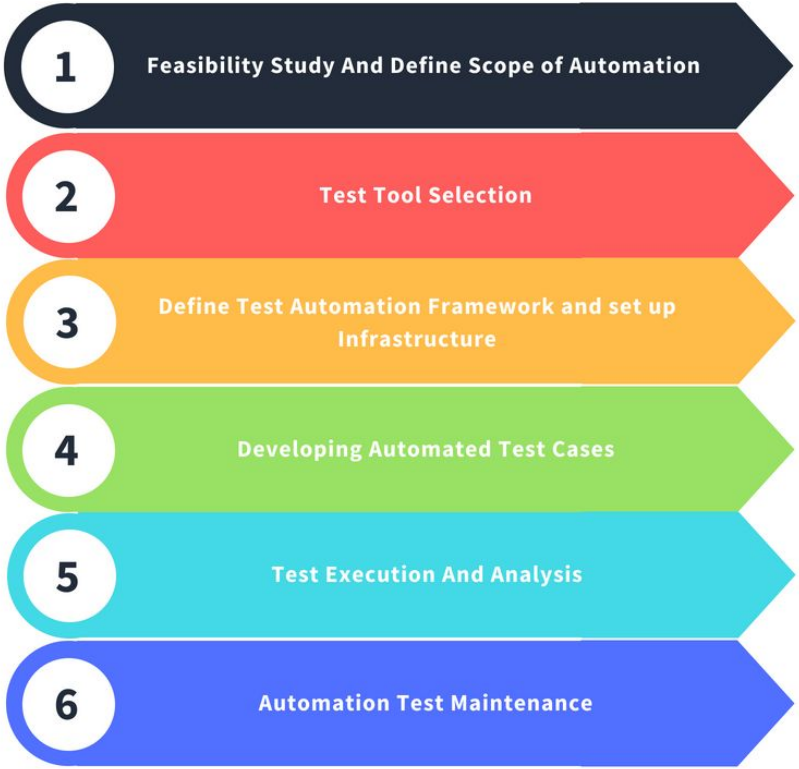
Types of Automation Tests

- **Functional Automated Tests**
  - Functional testing assures that the implementation of developers meet business requirements
  - Functional Automation testing assures that the functionality is automated
  - E.g for Functional Automation Tools: Selenium, UFT, RFT, VSTS
- **Non Functional Automated Tests**
  - The non functional testing is the type of testing done against the non functional requirements (performance, security, compliance)
  - Non functional automation assures that non functional parameters are automated (Tools: Jmeter, Load Runner)

Tools for Automation Testing

Licensed Tools	Open Source
1) QTP	1) Selenium
2) Win Runner	2) Jmeter
3) Test complete	3) Soap UI
4) Silk Test	4) Open STA
5) VSTS	
6) Tellurium	
7) Rational Functional Tester	

Tools for Automation Testing



When not to use Test Automation?

- **UI is changing frequently**: For applications still under development, or frequently changing UI, creating automated test scripts may be a waste of time
- **Subjective validation**: For application functions that require subjective validation such as usability, simplicity or look-and-feel, manual testing is more appropriate
- **Localization**: Testing localized content requires an understanding of the language, culture and local norms. These are best performed manually
- **One-timers**: The investment in developing test scripts pays of, if the test is repeated many times. It may not be worthwhile for one timers

Advantages of Automation Testing

- **Save Time**: This creates time for the tester to perform exploratory testing, concentrate on areas which cannot be automated and concentrate on other tasks.
- **Speed**: As automated tests are run by tools, these are run much faster than human users which adds to the first benefit of saving time
- **Repeatability**: The same tests can be rerun in exactly the same manner eliminating the risk of human errors
- **Reusable**: The automated tests can be reused on different versions of the software, even if the interface changes
- **Increased Coverage**: Testers can create a test suite with tests to cover every feature within the application

Limitations of Automation Testing

- **Need for scripting and programming skills:** Coding and technical skill level of resource should be good enough to write robust testing code
- **Need to maintenance of code**: Whenever application code is updated or modified, the code for automated test case must be also updated
- **Requires more initial development time**: Any new test automation would require time for development, creation of frameworks...
- **Increase tool needs**: Automation testing would increase the need for tools and also software required for the same
- **Complex analysis required when a test fails**: Is this test obsolete due to changes or does it have human mistakes?

Challenges of Automation Testing

- **Unrealistic expectations**: Generally there is a tendency to be optimistic about what can be achieved by a new test tool
- **Tool Limitations**: Too available in the market have drawbacks as well
- **Dependency on 3rd party integration**: Integrations with other applications, plugins, patches makes automation challenging
- **Lack of help and support required for the tool**: Some tools have extensive support required for the tool, others have to reply on internet and other user forums
- **Version compatibility for tool and browser:** The browsers are updated every rapidly in the market, but the tool supportability for the version make take considerable time

References

[1]. Kshirasagar Naik, Priyadarshi Tripathy, [2008], Software Testing and Quality Assurance, John Wiley & Sons, New Jersey.

[2]. Mauro Pezze, Michal Young, [2008], Software Testing and Analysis: Process, Principles, and Techniques, John Wiley & Sons, New Jersey.

[3]. Arnon Axelrod, [2018], Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects, Matan, Israel.

[4] Trần Cao Đệ, Nguyễn Công Danh, Giáo trình đảm bảo chất lượng phần mềm, NXB. Đại học Cần Thơ, 2014.