SOFTWARE TESTING ASSIGNMENT

MODULE – 5

DARSHAN GOHIL

WHAT IS AUTOMATION TESTING?

- Automation testing is a software testing technique that involves the use of specialized tools and scripts to automatically execute test cases, rather than relying on manual testing by a human.

- It is primarily used to speed up the testing process, increase test accuracy, and ensure consistency in the results. In automation testing, once test scripts are written, they can be reused multiple times across different versions of the application, making it highly efficient for repetitive and regression testing tasks.

- This type of testing is especially beneficial in large projects where frequent code changes require continuous validation.

- It plays a key role in agile and DevOps environments by enabling continuous testing and integration.

- Tools like Selenium, JUnit, TestNG, Appium, and Postman are widely used to automate web, mobile, and API testing.

- However, not all types of testing are suitable for automation.

- Tasks like exploratory testing, usability testing, and one-time test cases are better performed manually due to the need for human observation and judgment.

- Overall, automation testing helps improve software quality, reduces time and cost in the long run, and supports faster release cycles, especially when integrated into a continuous development and deployment pipeline.

WHICH ARE THE BROWSER SUPPORTED SELENIUM IDE?

- Selenium IDE (Integrated Development Environment) is a browser-based tool used for recording and playing back test cases.

- It supports automated testing primarily for web applications and is known for its simplicity and ease of use.

- Browsers Supported by Selenium IDE:

  - Google Chrome

  - Mozilla Firefox

  - Microsoft Edge *(via Web Extension support)*

WHAT ARE THE BENEFITS OF AUTOMATION TESTING?

➨ Automated tests run significantly faster than manual tests, especially for large test suites or repetitive tasks.

➨ Once created, automated test scripts can be reused across different versions of the application, saving time in future testing cycles.

➨ Automation allows more tests to be executed in less time, enabling wider coverage of features and scenarios, including edge cases.

➨ While the initial setup may be time-consuming and costly, automation reduces the need for manual testers, leading to long-term cost savings.

➨ Automated tests eliminate human errors that can occur during manual testing, ensuring consistent and reliable results.

➨ Automation is essential for modern DevOps practices, enabling seamless integration and deployment with tools like Jenkins, GitLab CI, and others.

➨ Automated tests are ideal for regression testing, ensuring that new changes don't break existing functionality.

➨ Testers can focus on exploratory and usability testing while automation handles repetitive and time-consuming test cases.

➨ Automation tools often generate logs and reports that make it easy to track defects, monitor performance, and analyze results.

WHICH ARE THE ADVANTAGES OF SELENIUM?

- Selenium is completely free and has a large community of users contributing to its improvement, making it accessible for individuals and organizations of all sizes.

- Selenium supports major browsers like Chrome, Firefox, Safari, Edge, and Opera, enabling cross-browser testing with ease.

- Test scripts in Selenium can be written using languages like Java, Python, C#, Ruby, JavaScript, and Kotlin, providing flexibility to testers.

- Selenium tests can be executed across different operating systems such as Windows, macOS, and Linux, allowing for true platform-independent testing.

- Selenium easily integrates with popular tools like TestNG, JUnit, Maven, Jenkins, Git, Docker, and CI/CD pipelines, enhancing test management and continuous testing.

- Selenium Grid allows you to run tests in parallel across multiple machines and browsers, saving significant time and resources.

- With Selenium, testers can create complex, customized test scenarios and frameworks tailored to project needs. It also supports behavior-driven frameworks like Cucumber.

- A strong user community, detailed documentation, and numerous tutorials make it easier to learn and troubleshoot issues.

- Selenium can run tests in headless mode using headless browsers like Headless Chrome or HTMLUnit, which speeds up test execution without launching the UI.

WHY TESTER SHOULD OPT FOR SELENIUM AND NOT QTP?

➡ Selenium is completely free and open-source, while QTP/UFT requires a paid license, which can be expensive for individuals and small teams.

➡ Selenium supports all major browsers like Chrome, Firefox, Safari, Edge, and Opera, and runs on Windows, macOS, and Linux.

➡ In contrast, QTP is limited to Windows and supports only Internet Explorer and Edge, making it less suitable for cross-platform testing.

➡ Selenium allows testers to write scripts in multiple languages such as Java, Python, C#, Ruby, JavaScript, etc.

➡ QTP primarily uses VBScript, which limits flexibility and integration with modern development tools.

➡ Selenium is designed specifically for web application testing, making it faster and more effective for modern web apps. QTP/UFT supports desktop and web apps but is heavier and slower for web-specific testing.

➡ Selenium integrates seamlessly with tools like Jenkins, Maven, Docker, and Git, making it ideal for continuous integration/continuous deployment (CI/CD) workflows. QTP has limited integration and is not well-suited for DevOps environments.