**ANSWER THE QUESTION BY GROUP 107**

**The question : Why and when do we use test automation?**

**Member 1 : Ngô Nguyễn Thanh Thanh**

Test automation is a critical practice in modern software development. Its primary purpose is to boost testing efficiency, precision, and speed, ultimately leading to improved software quality while saving both time and costs.

Test automation shines in various scenarios. Firstly, it excels at automating repetitive tests, allowing software teams to avoid the tedium of manually executing the same test cases repeatedly. This is particularly valuable in situations with a high volume of repetitive testing needs.

Regression testing, a key aspect of software maintenance, benefits significantly from test automation. Automated tests ensure that new code changes do not introduce unexpected issues into previously functioning components, providing confidence in software stability.

For teams adopting agile and continuous integration and delivery (CI/CD) practices, test automation is indispensable. It enables rapid feedback on code changes, aligning perfectly with the fast-paced, iterative development process.

Load testing, where software performance is assessed under various stress levels, is made more effective through automation. It helps identify performance bottlenecks and ensures a smooth user experience.

In complex scenarios, where interactions and processes are intricate, test automation offers systematic and error-free testing. Parallel execution capabilities further enhance efficiency, reducing the likelihood of human errors.

Integration into CI/CD pipelines ensures that only validated, high-quality code is integrated and deployed, maintaining software reliability.

Lastly, data-driven testing harnesses automation to evaluate diverse scenarios using various data inputs, bolstering test coverage and robustness.

In conclusion, test automation is a versatile tool that elevates software testing across multiple dimensions. By automating repetitive tasks, supporting swift regression testing, accommodating frequent releases, facilitating load testing, handling complex scenarios, enabling parallel execution, and seamlessly integrating with CI/CD, it contributes to enhanced software quality while optimizing resource allocation and development timelines.

**Member 1 : Trịnh Hoàng An**

Test automation is a critical practice employed in software development to improve efficiency and effectiveness in the testing process. We use test automation for several reasons. First and foremost, it significantly reduces the time required to execute repetitive test cases, enabling quicker feedback on code changes. This speed is essential in today's fast-paced development environments.

Secondly, test automation enhances test coverage. Automated tests can quickly cover a wide range of scenarios, ensuring that the software is thoroughly examined for bugs and defects. This comprehensive coverage is often unattainable through manual testing alone.

Furthermore, test automation contributes to cost savings over the long term. While initial setup and maintenance may require an investment, automated tests can be reused across multiple iterations and releases, reducing the need for extensive manual testing efforts.

We use test automation primarily when:

+ Regression Testing: To ensure that new code changes do not introduce regressions by re-running a suite of existing test cases.

+ Frequent Builds: In continuous integration and continuous delivery (CI/CD) pipelines, where automated tests are crucial to validate each code commit.

+ Load and Performance Testing: To simulate heavy user loads and analyze system performance under stress.

+ Cross-browser and Cross-platform Testing: To ensure compatibility across various environments.

+ Repetitive and Data-Driven Testing: For scenarios where the same test case needs to be executed with multiple data inputs.

* In conclusion, test automation is a vital practice used to save time, improve test coverage, and enhance software quality. Its application is most beneficial when integrated into CI/CD pipelines and when dealing with repetitive or resource-intensive testing scenarios.

Assignment table : z

|  |  |
| --- | --- |
| Trịnh Hoàng An | Ngô Nguyễn Thanh thanh |
| 1. Test Automation Engineer 2. Quality Assurance (QA) Team 3. Project Manager | 1. Test Automation Engineer  2. Quality Assurance (QA) Team  3. Development Team |

1. **Test Automation Engineer**: This is the individual primarily responsible for creating, maintaining, and managing the test automation project. They should have expertise in test automation tools, frameworks, and scripting languages. Their role includes designing test cases, developing automation scripts, and ensuring the overall health of the automation suite.
2. **Quality Assurance (QA)** : The QA team plays a crucial role in collaborating with the Test Automation Engineer. They define test cases, provide test data, and validate the results of automated tests. They also report and verify defects identified during automation testing.
3. **Development** : Developers may contribute to the automation project by providing insights into application architecture, code changes, and best practices for integrating automated testing into the development process.
4. **Project Manager**: The project manager oversees the overall test automation initiative. They allocate resources, set priorities, and ensure that the automation project aligns with the broader project goals and timelines.