ESSAY QUESTIONS

1.

1. Automation tool to use

    a. I would use Selenium for my web application automation testing.

Web application testing can be automated with the help of Selenium, a popular open-source automation tool. Developers and testers may use it to efficiently automate the manual testing process and run tests on many operating systems and browsers.

I would recommend using Selenium because to its cross-browser interoperability, which makes it simple to test your web application on various browsers like Chrome, Firefox, Safari, and Internet Explorer and confirm that it functions as intended across various operating systems and browsers.

Additionally, it enables you to create scripts that can be used repeatedly to test various aspects of your program. You may concentrate on other important testing areas as a result of the time and effort savings.

In order to set up a complete testing environment, Selenium may be readily connected with additional testing tools like TestNG, JUnit, and Jenkins. The CI/CD process can really benefit from this.

Last but not least, it is ideal for use by all kinds of individuals and enterprises due to its open source nature and extensive online community for support.

    b. I would use Appium for my mobile application automation testing.

With the help of the open-source automation tool Appium, software testers can fully automate the testing of mobile applications on both Android and iOS.

Appium is the best option for testing mobile applications because it is compatible with both the Android and iOS operating systems. Using Appium, you can test your application on various hardware and software configurations to make sure it performs as intended on all of them.

Developers and testers can quickly build scripts in the language of their choice and integrate Appium into their current development workflow as it supports many programming languages.

Appium enables you to test your mobile application on actual devices, giving you a more accurate idea of how it will function in practice.

Setting up a full testing environment is made simple by the ease with which Appium can be coupled with well-liked testing frameworks like TestNG, JUnit, and Cucumber.

Appium is a free to use open-source technology with a big developer community that contributes to its growth and maintenance.

Other testing frameworks that could as well help achieve the same goals that I personally like are **Splinter** and **Pyppeteer**. Splinter is a Python library that provides a simple and intuitive interface for automating browser actions, while Pyppeteer is a Python library that provides a high-level API for controlling headless Chrome or Chromium browser.

I have as well found myself using **Robot framework** for most of these tests and using the Libraries for Selenium, Appium, Splinter, Pyppeteer for Web, Mobile and Desktop application testing. I also use it mostly for API testing, Database testing, Web Scrapping and any other automation task I need to perform.

2. How we ensure we are testing on all devices

Perform a market analysis to identify the gadgets that are most popular with your target audience. The most popular devices and operating systems can be determined by looking at market share reports, consumer surveys, and analytics data.

Make a device matrix: List all the hardware and operating systems you need to test on your device matrix. The operating system version, device type, manufacturer, and model should all be listed in the device matrix.

Devices should be prioritized according to their market share, customer feedback, and analytics data. Setting device priorities can help you efficiently use your testing resources and guarantee that you are testing on the most pertinent devices.

Utilize device emulators and simulators: Use these tools to test on gadgets you don't physically have access to. To test on various screen sizes, resolutions, and operating system versions, emulators and simulators can be utilized.

Crowdtest: Use services for crowdtesting to test on a variety of devices. Crowdtesting services give you access to a sizable pool of actual testers who can evaluate your application across a range of devices.

Continuous observation: Keep an eye on the gadgets your consumers are utilizing. Consider adding new hardware and operating system versions to your testing matrix by analyzing user input and analytics data.

2.

I would perform a Regression testing in this case.

Regression Testing is a type of testing that is done to verify that a code change in the software does not impact the existing functionality of the product.

This is to ensure that the product works fine with new functionality, bug fixes or any changes to the existing feature.

It is part of the pre-deployment testing that aims to find underlying bugs and issues so they can't make their way into production. Performing end-to-end testing, verifying testing, and manually verifying features and functionalities is part of the pre-deployment testing process.