**1.Have you ever done with mobile testing?**

Yes, I have experience in mobile testing on both Android and iOS platforms. Over the past year, I've performed functional and non-functional testing, including usability, performance, and security testing. I wrote and executed detailed test cases, and reported bugs using tools like JIRA and Appium. My work involved both manual testing and using automated testing tools for regression tests.

**2.How many and what kind of devices did you test before?**

I've tested on a wide range of devices to ensure broad compatibility. For Android, I tested on devices like Samsung Galaxy, Google Pixel, and OnePlus, covering different screen sizes and OS versions. For iOS, I tested on various iPhones from iPhone 6 to iPhone 12, as well as iPads. I also used Android emulators and iOS simulators for initial testing phases to quickly identify issues before moving to real device testing.

**3.What different between mobile testing and web testing?(Hường)**

### **First, regarding the versions tested:**

### With the diversity of mobile devices (smartphones, tablets, iPads, etc.) and operating systems (iOS, Android, Windows, Blackberry, etc.), compared to the relatively small number of web browsers, testing on mobile applications is a significant challenge for testers.

### **Second, screen size:**

When testing websites, we are usually familiar with PC screens with relatively uniform sizes. However, with mobile testing, due to user preferences and competition among providers, the screen sizes of mobile devices are generally diverse and constantly changing.

When performing tests, especially interface tests, testers need to pay attention to cases with different screen sizes to check for layout issues, and whether button sizes, text boxes, radio buttons, etc., change or cause difficulties for users.

**Third, user interaction:**

On websites, users usually interact with the system through the keyboard and mouse. Interaction with other devices is possible but less common as PCs generally do not have built-in cameras or microphones.

Conversely, with mobile apps, users interact in various ways: touch, swipe, rotate, drag, drop, voice, Face ID, etc. Therefore, testers need to understand these cases to test the system more thoroughly and comprehensively.

**Fourth, user experience:**

Developers usually focus on whether a function works correctly. However, testers play an important role in experiencing the application from a user's perspective to provide the best feedback and adjustments.

**Fifth, user security and privacy:**

On mobile devices, test cases related to access permissions and privacy arise. For example, permissions to access the camera or microphone.

**Sixth, network connection:**

When testing web applications, we primarily test under successful network connection conditions. Cases involving fast or slow network connections for applications are rarely addressed.

In mobile testing, most devices require a network connection. Applications operating with 3G, 4G, 5G, varying signal strengths, signal loss, or users moving at different speeds can all affect the application's operation. Therefore, covering these cases during testing ensures a better user experience.

**Seventh, cases of installing, uninstalling, upgrading apps, and upgrading OS versions:**

We need to note cases such as:

* Can the app be installed or uninstalled after updating the OS version?
* Does the user’s data get lost after upgrading the app version?
* Can the app still be used if the app version is not upgraded to the latest?

**Eighth, how does the application perform when interrupted?**

On mobile devices, users often use multiple applications simultaneously, such as SMS and incoming calls. How does the app operate in such interrupt scenarios? Can it save ongoing actions? These cases are less common on web applications.

**4.How to choose the device and version to execute test?**

I choose devices based on market share to cover popular models, ensure a mix of different screen sizes and hardware, test on both latest and older OS versions, follow client requirements, and use risk-based testing to focus on high-risk areas.