1. **Why do you need test case?(Hoàng Hường)**

A test case suite acts like a checklist. Each time testing is performed, it is based on this checklist, ensuring no cases are missed. Additionally, it facilitates the handover of work when I am no longer involved in the project and it is handed over to another tester. The new person will rely on the test cases to continue their work. Therefore, building an effective test case is extremely important. When writing, I must ensure that I do not "assume" anything. The test cases must be detailed, accurate, and continuously updated, as not everyone has the same background and understanding of the system as I do.

Having a test case suite makes creating test runs easier, allowing me to track the software development process and the project's progress. A test case suite also helps in tracking bugs more effectively. With a checklist of all the cases to be tested, it becomes easier to filter out the bugs present in the software.

From the test case suite, I can determine whether the test case coverage is adequate. If I have executed all the test cases and the software still has bugs, it means that the test cases are not well-written. Whenever a bug arises and the flow is not covered in the test case, it needs to be added immediately. No one can create a complete test case suite from the beginning; test cases will be supplemented throughout the software development process.

**2. Test case format?(Hoàng Hường)**

A test case format typically includes several key components:

* **Test Case ID** :To uniquely identify each test case
* **Category:** classify and organize test cases based on their characteristics or the area of the application they cover
* **Test Case Title/Description** :A detailed explanation of what the test case is designed to test.
* **Pre-conditions:**Any prerequisites or conditions that must be met before executing the test case
* **Test Steps:** A step-by-step guide on how to execute the test case
* **Test data:** Data required for executing the test case.
* **Expected Result:** The expected outcome if the test case passes.
* **Status:** Indicates whether the test case has passed or failed based on the comparison between expected and actual results.
* **Notes:**Any additional information, observations, or notes relevant to the test case
* **Prority (optional)**:The importance and urgency of the test case.

**3.What is test case priority? Why do you need this field?(Hoàng Hường)**

* Test case priority is a classification that indicates the importance and urgency of a test case within the testing process. It helps testers determine the order in which test cases should be executed based on their significance and the potential impact on the overall project.
* Test case prioritization makes test execution effective and increases the chances of finding and fixing bugs early on in the testing stage. It allows you to run the critical test cases before others so that defects can be identified. This helps save costs related to fixing defects by up to 85%.

**4. What are best practices for writing test cases? (Tùng Anh)**

To write effective test cases, first, understand the requirements and acceptance criteria. Each test case should have a specific purpose, written in clear and detailed language, including steps to perform, preconditions, inputs, actions, and expected results. Ensure test cases cover all scenarios, including positive, negative, edge cases, and boundary conditions. Test cases should be linked to requirements or user stories to maintain traceability. Prioritize executing important test cases first and reuse existing test cases for similar functionalities. Finally, each test case should be written so that it can be executed independently.

**5. How many test cases can you execute/write in a day?(Tùng Anh)**

The number of test cases a tester can execute or write in a day depends on factors such as the complexity of the test cases, the readiness of the test environment and data, and whether the tests are automated or manual. On average, a tester might execute or write between 15 to 30 simple test cases per day, but this number can vary widely based on the specific circumstances.

**6. What is test data?(Bảo Trân Jenny)**

Test Data in Software Testing is the input given to a software program during test execution. It represents data that affects or affected by software execution while testing. Test data is used for both positive testing to verify that functions produce expected results for given inputs and for negative testing to test software ability to handle unusual, exceptional or unexpected inputs.

**7. How many test design technique do you know? Short describe and**

**example?(Bảo Trân Jenny)**

* Equivalence Partitioning: Divides input data into partitions of valid and invalid values. Test cases are then chosen from each partition.

Example: For an age input field that accepts values between 18 and 65, test cases might include 17 (invalid), 18 (valid), 65 (valid) and 66 (invalid).

* Boundary Value Analysis: Tests the boundaries between partitions of input values. Focuses on testing values at the edges of these partitions.

Example: For the same age input field (18 to 65), test cases would include 17 (boundary - invalid), 18 (boundary - valid), 19 (boundary - valid) and 65 (boundary - valid), 64 (boudary - valid), 66(boundary - invalid).