## Questionnaire:

Please answer the following questions. Include your responses to the same document where you collect findings from practical testing task.

1. Are you used to also checking console logs and network traffic? If there is a need to manipulate network requests, what is your favorite tool for that job?
2. Please review this scenario: We have a mini evaluation which consists of 5 questions. Each question can be answered correctly or incorrectly. We need to calculate the amount of ALL possible cases (for example, the user answers correctly on the first 3 questions and incorrectly on the next 2, etc). Please write down the amount of all possible cases (the number). Explanation is optional.
3. Imagine you are testing a new feature in the App or Web that uses data loaded from API. There's only a minimal amount of example data present in DB at the moment and back-end developers are busy, so they can't generate more now, only the next day. What will be your next steps? Please describe all options, if you have multiple.
4. Please describe what are the specifics/challenges of Mobile Apps testing in your opinion?
5. In your opinion, when is it reasonable to use a checklist and when it's more reasonable to have test cases?
6. Yes, I'm accustomed to checking console logs and monitoring network traffic. As for manipulating network requests, my favorite tool for that is Burp Suite.
7. (32).
8. -Utilizing the minimal example data available to conduct basic testing of the new feature.

-Creating my own test data or generating large datasets for more thorough testing.

-Considering the option of using API simulation tools such as Mock servers or other tools to create temporary APIs for testing purposes.

-Rescheduling tests and the testing cycle to accommodate the availability of additional data on the next day.

-Reaching out to developers to discuss the possibility of expediting the process of obtaining additional test data or exploring alternative avenues.

1. -Device and platform diversity: Mobile applications need to work across different devices (Android, iOS) and versions of operating systems, which can lead to compatibility issues and display discrepancies.

-OS versions and updates: Ensuring the compatibility of the application with various versions of operating systems and providing updates to support new features and bug fixes is crucial.

-Screen variations: Mobile devices come in various screen sizes and orientations, which can affect the display and interaction with the application.

-Resource constraints: Mobile devices have limited resources such as memory, power, and processing power, which can affect the performance and execution of the application.

-Network interaction: Testing under different network conditions (WiFi, mobile data) and handling connection restoration after network disruptions.

-Security: It's essential to verify user data protection, restrict access to sensitive information, and manage permissions.

1. Checklists are typically useful for testing routine or repetitive tasks where it's important to track the execution of a specific set of actions without the need for a detailed description of each step. They can be an effective tool for verifying compliance with basic standards and quality requirements. Checklists allow for quick verification of the completion of mandatory steps or conditions without the need for deep detailing.

Test cases are usually used for more detailed testing where it's necessary to precisely describe each test step, expected results, and acceptance criteria. They are effective for complex functional testing or testing that requires close attention to detail. Test cases help to systematize the testing process and provide greater structure in executing test scenarios.