

1. Elaborate on the differences between test cases and test plans with relevant examples.

Test plan:

A test plan is a high-level document that describes the overall strategy for testing a product. It specifies what will be tested, the way the testing is going to take place, the tools that will be used, who will be carrying out the testing, and what are the risks that may arise. In simple terms, a test plan acts as a guideline that drives the whole testing process in the beginning to the end.

For example, in the case of a new headphone the test plan will state that the team will test these features microphone, volume buttons, sound quality, connectivity and battery performance. It will also contain the schedule of the testing, tasks of each tester, and any problem that that may affect testing.

Test case:

A test case is a detailed and step-by-step instruction that is used to verify one particular functionality of the product. A test case contains the test case id, test objective, test steps, expected result, actual result, and the status (pass or fail). The purpose of test cases is to help testers test in the same clear and easy way every time.

For example, a test case may only test the defined button, which is the volume up button, by pointing out simple tests as follows: turn on the headphone, press the volume-up button and confirm whether the volume is increased. This shows the difference clearly.

3. Prepare a test plan for the above headphone produced.

Test Plan for Headphone with Detachable Microphone

1. Introduction

This test plan explains how I will test a new headphone that has a detachable microphone, volume buttons, and a mic ON/OFF switch. My goal is to check whether the headphone works properly and has no major problems.

2. What I Want to Test

1. If the microphone works only after it is connected
2. If the volume buttons work correctly
3. If the mic ON/OFF switch works
4. If the audio coming from the headphone is clear
5. If the left and right sound channels work properly
6. If the headphone works on different devices
7. If wired/Bluetooth connection is stable

3. What Is Included (Scope)

Things I will test

1. Physical condition of the headphone
2. Microphone detection and voice quality
3. Buttons (Volume Up, Volume Down, Mic ON/OFF)
4. Sound quality and channel check
5. Call audio
6. Basic durability (normal button pressing)

Things I will NOT test

1. Water resistance
2. Long-term durability
3. Battery testing (if wired model)

4. Test Items

1. Headphone
2. Detachable microphone
3. Volume buttons
4. Mic ON/OFF switch
5. Audio jack/Bluetooth (depending on model)

5. How I Will Test

1. I will perform manual testing
2. I will connect the headphone to phone/laptop
3. I will play music and make calls
4. I will press each button and check the response
5. I will test the mic ON/OFF behavior
6. I will test the headphone on different devices
7. I will also do negative tests (like trying to use mic without attaching it)

6. Test Environment

1. Android phone
2. iPhone (if available)
3. Laptop
4. Voice recorder app
5. Music player
6. Quiet room for mic testing

7. When Testing Will Start

1. Headphone and microphone are available
2. Devices are ready
3. Test cases are prepared

8. When Testing Will Finish

1. All test cases completed
2. Failed cases reported
3. Major issues fixed and retested

9. Risks

1. A defective unit may affect results
2. Background noise may affect mic quality test
3. Some devices may have compatibility differences

10. Final Output (Deliverables)

1. Test Plan
2. Test Cases
3. Bug Report (for failed test cases)
4. Final summary of results

4. Difference Between Bug and Issue with Bug Life Cycle

Difference Between Bug and Issue

In software testing, the terms bug and issue are related, but they do not mean the same thing.

Bug

A bug is a defect or error in the product that causes it to behave incorrectly. It happens when the actual result is different from the expected result.

Example:

- The Volume Up button does not increase the sound.
- The microphone is connected but still does not work.
- These are bugs because they affect the functionality of the product.

Issue

An issue is a broader problem that can affect the project, but it is not always a defect in the product.

Issues can include:

- Bugs
- Missing requirements
- Testing delays
- Device not available
- Network or system problems
- Miscommunication in the team

So, a bug is always an issue, but an issue is not always a bug.

Bug Life Cycle

The bug life cycle shows how a bug moves from the moment a tester finds it until it is completely fixed.

1. New

The tester finds a problem and creates a bug report.

Status: New

2. Assigned

The QA Lead or Team Lead assigns the bug to a developer.

Status: Assigned

3. Open

The developer starts working on the bug and investigates the cause.

Status: Open

4. In Progress / Fixed

The developer fixes the bug and updates the status.

Status: Fixed

5. Retest

The tester tests the bug again to confirm whether the fix is working correctly.

Status: Retest

6. Verified

If the tester finds that the bug is truly fixed, they mark it as verified.

Status: Verified

7. Closed

When everything works properly, the tester closes the bug.

Status: Closed

8. Reopened (If needed)

If the bug still appears even after the fix, the tester reopens it.

Status: Reopened

Bug Life Cycle:

New → Assigned → Open → Fixed → Retest → Verified → Closed
(or Reopened if it still exists)

6. Explain briefly about UI/UX Tests and prepare a Checklist for UI/UX Testing.

What is UI Testing?

UI Testing (User Interface Testing) focuses on how the product looks.

It checks the visual elements of an app, website, or device to make sure everything is displayed correctly.

In UI testing, we check:

- Buttons
- Icons
- Text size
- Colors
- Alignments
- Layout
- Spacing between elements

Simple example:

If a button is cut off or text is too small, it fails UI testing.

What is UX Testing?

UX Testing (User Experience Testing) focuses on how the product feels when used.

It checks whether the user can use the product easily, without confusion or difficulty.

In UX testing, we check:

- Is the product easy to use?
- Are the controls understandable?
- Does the user get confused anywhere?
- Is the product comfortable and smooth to use?

Simple example:

If the mic ON/OFF switch is hard to find or the button is too stiff, it fails UX testing.

UI/UX Checklist

Below is a complete checklist that you can use for your assignment.

UI Testing Checklist

| UI Test Item | Checklist Status |
|--|--------------------------|
| Buttons are clearly visible | <input type="checkbox"/> |
| Icons are properly displayed | <input type="checkbox"/> |
| Text is readable and not too small | <input type="checkbox"/> |
| Colors are consistent and not confusing | <input type="checkbox"/> |
| Elements are properly aligned | <input type="checkbox"/> |
| No overlapping text or broken UI | <input type="checkbox"/> |
| Labels (e.g., Volume, Mic) are clearly written | <input type="checkbox"/> |
| Indicators (LED, ON/OFF) display correctly | <input type="checkbox"/> |
| Spacing between elements is neat | <input type="checkbox"/> |
| All UI components follow the design standard | <input type="checkbox"/> |

UX Testing Checklist

| UX Test Item | Checklist Status |
|--|--------------------------|
| Headphone buttons are easy to reach | <input type="checkbox"/> |
| Mic attachment feels simple and smooth | <input type="checkbox"/> |
| Buttons work without applying too much force | <input type="checkbox"/> |
| User understands which button does what | <input type="checkbox"/> |
| Volume control feels natural to use | <input type="checkbox"/> |
| Mic ON/OFF switch is easy to locate | <input type="checkbox"/> |
| No confusion while using the product | <input type="checkbox"/> |
| Wearing the headphone feels comfortable | <input type="checkbox"/> |

| UX Test Item | Checklist Status |
|---|--------------------------|
| All functions work in a logical flow | <input type="checkbox"/> |
| Overall experience feels smooth and user-friendly | <input type="checkbox"/> |

7. If you encounter a technology or tool you're not familiar with during an internship project, how would you go about learning it?

When I encounter a technology or tool that I am not familiar with during an internship, I follow a simple and effective learning process. First, I start by doing basic research to understand what the tool is and why it is used. Then I read the official documentation because it explains the features clearly and teaches the correct way to use the tool. After that, I watch tutorials or videos to see practical examples, which helps me understand the steps visually. I also practice the tool by trying small tasks on my own before using it in the real project. If I still have questions, I ask seniors or teammates for help, as they can guide me based on their experience. While learning, I take simple notes so I can remember the steps later. Finally, I start applying the tool in the actual project, because real practice helps me learn the fastest. This is how I learn any new tool confidently during an internship.

8. You can provide your personal experience here that motivates you for this work, which is why you selected QA for the internship.

My interest in Quality Assurance began when I was working on my personal project, the **Chee Game App**. While developing the game, I spent a lot of time testing different features such as buttons, game logic, scoring, and animations. During this process, I found many small mistakes like buttons not responding, scores not updating correctly, and unexpected crashes. Finding these issues and fixing them helped me understand how important proper testing is before releasing any product. I also enjoyed looking at the app from a user's point of view and making sure everything worked smoothly.

This experience motivated me to choose QA for my internship. I realized that I naturally pay attention to details, enjoy solving problems, and like improving the overall user experience. QA allows me to work with developers, understand how things work, and make sure that the final

product is stable and user-friendly. Working on my Chee Game App showed me that testing is not just about finding errors, it is about making the product better. That personal experience encouraged me to build my skills and pursue an internship in QA.