**Mobile App Testing Strategy – Demoblaze**

If Demoblaze app also has a mobile version for Android and iOS, since the build isn’t available yet, I’ve created a test plan that can be executed once the APK is ready.  
This plan mainly focuses on the user journey, app behavior under different conditions, and mobile-specific features like push notifications and deep links.

Before starting mobile app testing, it’s important to clearly understand the requirements and features expected in the app.  
For the Demoblaze mobile version, the requirements will be gathered from multiple sources such as:

* The web version (as a reference for expected functionality)
* API documentation for backend behavior
* Product requirement documents (PRDs**)** or user stories shared by the product team
* Discussions with developers and business analysts to clarify any assumptions

**Approach**

Testing will include both functional and exploratory checks to make sure all main flows work smoothly.  
I’ll use a mix of emulators and real devices to test compatibility, performance, and notifications.

**Main focus areas:**

* Login, product search, add to cart, and checkout flows
* Push notifications and deep link handling
* Performance in different network conditions
* Accessibility and general user experience(touch/scroll/click)
* Testing on the devices with android OS versions 10-14 or from oreo to latest version and ios versions 15-17

**Functional & UX Checks**

1.Signup: Verify with valid, invalid, and already registered user data.

2.Login and Logout: Check valid and invalid credentials, error messages, and proper session handling.

3.Product Listing & Search: Validate product filters, sorting options, and smooth navigation between screens.

4.Cart Operations: Add and remove products, update quantity or price, and verify cart data persists after relaunch.

5.Checkout: Test complete flow — address selection, payment gateway behavior, success and failure handling.

6.Deep Links: Ensure product, offer, and order-specific links open the correct screen in the app.

7.Screen Responsiveness & Readability: Check if the layout adjusts correctly on different screen sizes and orientations.

8.Smooth Scrolling: Ensure product lists and pages scroll smoothly without lag.

9.Screenshot Handling: Confirm that screenshots taken during sensitive actions (like payments or personal info) do not expose confidential data.

10.Push Notifications: Verify that tapping on notifications opens the correct screen, and messages are displayed accurately.

**Network & Accessibility**

* Test the app on offline, 3G/4G/5G, and Wi-Fi connections.
* Switch between networks (Wi-Fi ↔ mobile data) during checkout and verify the app handles it gracefully.
* Check TalkBack (Android) and VoiceOver (iOS) to ensure all UI elements are readable and accessible to visually impaired users.
* Validate font scaling, color contrast, and touch target sizes for accessibility compliance.
* Test Voice Search (Android) to verify that spoken input in the search bar works correctly and brings accurate product results.
* Check Siri integration (iOS) to ensure voice commands like “Open Demoblaze” or “Search for phones” open the correct app section.

**Test Case Matrix for extended features**



**Automation Plan**

I’ll automate Login the below (main smoke) flows using Appium with Python

* Search → Add to Cart → Checkout
* Deep link and push notification validation
* Offline/online transition scenarios

Testing can be done on **BrowserStack** to cover multiple devices and OS versions.

**Summary**

This plan covers how I’ll test the Demoblaze mobile app once it’s ready.Push notifications and deep links are critical because they directly affect user engagement and navigation.By testing these carefully, we can ensure users always land on the correct screen and have a smooth experience, whether they’re online, offline, or switching networks.