LinkedIn Mobile Automation Test Plan

1. Introduction

This document outlines the strategy and approach for automated testing of the LinkedIn mobile app (Android). The goal is to ensure LinkedIn's app delivers a high-quality, reliable, and seamless user experience by validating features across mobile platforms, devices, and user scenarios.

2. In Scope

- Mobile automation testing for LinkedIn features: login, profile management, messaging, feed browsing, connections, and searching.
- Automation of workflow validation, UI/UX checks, functionality.

3. Out of Scope

All types of testing exclude Functionality Testing, like:

- Non-mobile features (web functionality).
- API Testing.
-Database Testing.

4. Test Schedule

Feature	Responsible QA	Milestone/Release Target
Login/Auth	Ahmed	Sprint 1
Feed Browsing	Aya Sh	Sprint 2
Messaging	Aya Ali	Sprint 2
Profile/Edit	Ahmed	Sprint 3
connections	Haya	Sprint 3
Searching	Haya	Sprint 4

5. Test Approach

- Agile methodology with daily iterations and rapid feedback.
- Automation using Appium (Android).
- Functional, UI, and regression for core flows.
- Test cases assert navigation, data correctness, device compatibility, app behavior, and performance.

6. Suspension Criteria and Resumption Requirements

-Suspension: Testing will be suspended if there is an internet connectivity issue affecting the Gaza side, interrupting access to the LinkedIn mobile app or test infrastructure.
- Resumption: Testing will resume once internet connectivity in Gaza is restored and stable enough to support normal testing operations.

7. Milestones and Deliverables

Before Testing: mobile user stories, SRS alignment, app builds, test plan, test suite.
During Testing: automated execution, result logging, defect tracking.
After Testing: test summary, pass/fail metrics.

8. Responsibilities

- Project Manager: resource, timeline, and risk oversight.
- Scrum Master: facilitate ceremonies and resolve blockers.
- QA Team: writes/executes test cases, prepares test data, defect reports, review meetings, and device coverage.

9. Test Environment

- OS: Android.
- Devices: smartphones.
- Tools: Appium, emulator/simulator, and Elcipes.