🧪 Test Summary Report – Flipkart Automation Project

# 1. Project Overview

**Project Name**: Flipkart Automation Testing  
Test Type: Functional UI Automation  
Tools Used: Selenium WebDriver (Python), PyTest, PyTest-HTML  
Automation Framework: PyTest + POM (Page Object Model)

**Test Environment:**- OS: Windows 10  
- Browser: Google Chrome v117+  
- Language: Python 3.11  
- Resolution: 1366x768

# 2. Test Modules Covered

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Module | Test Scenario | Status |
| 1 | Login | Valid mobile number login | ✅ Pass |
| 2 | Login | Blank mobile number | ✅ Pass |
| 3 | Login | Invalid mobile number | ✅ Pass |
| 4 | Search | Search for a product | ✅ Pass |
| 5 | Filter | Apply brand & price filters | ✅ Pass |
| 6 | Cart | Add product to cart | ✅ Pass |
| 7 | Cart | Empty cart behavior | ✅ Pass |
| 8 | Buy Now | Click without login | ✅ Pass |

# 3. Test Execution Summary

- Total Test Cases: 15  
- Passed: 15  
- Failed: 0  
- Execution Time: ~366.99 seconds  
- Report: report.html (generated by PyTest)

# 4. Screenshots & Reports

- Screenshots captured on failure (folder: /screenshots)  
- HTML Report generated via:  
 pytest tests/ --html=report.html

# 5. Known Issues / Limitations

- OTP entry not automated (Flipkart sends real OTP)  
- No real purchases made (Buy Now stops at login)  
- Dynamic element delays managed with static waits (time.sleep() used temporarily)

# 6. Conclusion

The automation suite for Flipkart core flows has been implemented successfully using Selenium with PyTest and POM architecture. The results indicate all critical UI flows are stable under current test conditions.