STLC & Test Planning Documentation

# Project: Online Food Delivery App

## 1. Functional vs Non-Functional Requirements

### Functional Requirements:

|  |  |
| --- | --- |
| ID | Requirement |
| FR1 | User registration and login |
| FR2 | Restaurant and menu browsing |
| FR3 | Add to cart and place orders |
| FR4 | Online payment integration |
| FR5 | Order tracking and delivery updates |
| FR6 | User feedback and ratings |

### Non-Functional Requirements:

|  |  |
| --- | --- |
| ID | Requirement |
| NFR1 | System should load pages within 2 seconds |
| NFR2 | App must handle 10,000 concurrent users |
| NFR3 | Payment data must be encrypted |
| NFR4 | 99.9% uptime availability |
| NFR5 | Responsive UI across all devices |

## 2. STLC Phases for Online Food Delivery App

|  |  |
| --- | --- |
| Phase | Description |
| Requirement Analysis | Understand user and system requirements (functional + non-functional) |
| Test Planning | Define test objectives, strategy, resources, tools, schedule |
| Test Case Design | Create test cases for login, ordering, payment, tracking, etc. |
| Test Environment Setup | Set up mobile devices, backend servers, databases |
| Test Execution | Execute test cases, log defects, perform regression testing |
| Test Closure | Ensure all tests are complete, generate test summary report |

## 3. Test Strategy (in your own words)

Our test strategy will involve both manual and automated testing.  
We will perform:  
- Functional Testing to check core features (login, ordering, payments)  
- Smoke Testing to verify build stability  
- Regression Testing after bug fixes  
- Performance Testing using tools like JMeter  
- Security Testing for payment and login  
- Cross-platform Testing on Android/iOS/web

## 4. Sample Test Plan

### Test Plan Overview

Objective: To ensure the Online Food Delivery App meets business and user requirements without defects.

Scope: Testing of all major modules including registration, order placement, payment, and notifications.

### Entry Criteria

- Approved business requirements document  
- Application build ready for testing  
- Test environment setup complete

### Exit Criteria

- All critical test cases passed  
- No critical/severe bugs open  
- Test summary report created

### Tools Used

- JIRA – Bug tracking  
- TestRail – Test case management  
- Selenium/Appium – Test automation  
- Postman – API testing