

Software Test Plan (STP) – Helping Hands

Project: Helping Hands software

Version: 1.0

Authors: <QA Team>

Date: 31-08-2025

Status: Sample / Draft

1. Introduction

The purpose of this Test Plan is to define the testing strategy, scope, resources, schedule, and deliverables for the Helping Hands Software (HHS). The software enables company employees to volunteer at marts during weekends/festivals, manage slot bookings, award HHP points, and allows Store Admins and Main Admins to manage operations. This Test Plan ensures that the system meets all functional and non-functional requirements as defined in the SRS.

2. Test Items

The following components/modules will be tested:

- User Registration & Login
- Slot Search & Booking
- Slot Cancellation (max 2/day)
- Waitlist Mechanism (FCFS)
- Store Details Management
- Section/Requirement Management
- Attendance Module
- HHP Points Calculation
- Notifications (SMS/Email)
- User, Store, and Performance Management (Admin Panel)

3. Features to be Tested

- Location-based mart search
- Slot booking with FCFS rules
- Waitlist assignment and promotion
- Cancellation restrictions enforcement
- Store Admin's ability to modify store details
- Attendance marking and HHP update
- Main Admin's user/store management
- Notification delivery
- UI responsiveness & validation
- Security features: RBAC, authentication, encryption

4. Features Not to be Tested

- Integration with real payment gateways (none used)
- Third-party SMS/Email vendor internal reliability
- Actual GPS hardware accuracy (only API responses tested)
- Multi-language UI support (not in scope)
- Performance testing beyond 1000 users unless required

5. Test Approach / Strategy

Levels: Unit Testing, Integration Testing, System Testing, User Acceptance Testing

Types: Functional, UI/UX, Security, Performance, Compatibility, Regression

Entry Criteria: Code completed, environment ready, test cases prepared

Exit Criteria: 95% test cases passed, no critical defects

5.1 Security Validation

- Test RBAC for User, Store Admin, Main Admin
- Validate secure login & JWT usage
- Verify HTTPS enforcement
- Ensure location permission is opt-in only
- Validate encryption of sensitive data

6. Test Environment

Hardware: Windows 10/11 PC with 8GB RAM , Android/iOS mobile devices for responsive testing

Software: Node.js backend, MongoDB database , React frontend ,Browsers: Chrome, Firefox, Edge

Tools: Postman (API testing), Selenium / Cypress (UI automation), JMeter (performance testing), GitHub (version control)

Test Data: Dummy users (Users, Store Admins, Main Admins), Sample stores, sections, slots, Predefined booking scenarios (empty, full, waitlist)

7. Test Deliverables

- Test Plan (this document)
- Test Cases (manual & automated)
- Test Scripts
- Test Data
- Test Execution Logs
- Defect Reports
- Test Summary Report

9. Roles and Responsibilities

Role	Name	Responsibility
QA Lead	K Sriharsha	Prepare plan, coordinate execution
Test Engineer	Kushal Kumar G	Design & execute test cases, log defects
Developer	K Sreekar	Support defect fixes and triage
Product Owner	Kirthan G	Approve test results, sign-off readiness

10. Risks and Mitigation

Risk	Mitigation
High load during peak days	Conduct load testing & enable backend scaling
SMS/Email failures	Implement retry logic + fallback email
Incorrect HHP calculation	Double validation & test with boundary inputs

Risk	Mitigation
Role misuse by admin	Enable audit logs for admin actions
Data loss	Automated backups & recovery verification

11. Assumptions & Dependencies

- Users have stable internet access
- Third-party SMS/Email APIs remain available
- GPS/Geolocation services functioning normally
- Test environment mirrors production behaviour

12. Suspension & Resumption Criteria

Suspend testing if:

- High severity issues block major workflows
- Test environment becomes unavailable
- Backend/API failure rate becomes critical

Resume testing if:

- Blocking issues are fixed
- Environment restored
- Stability validated via smoke testing

13. Test Case Management & Traceability

- All requirements traced to test cases through RTM
- Test cases mapped to FRs, NFRs, SRs
- Every defect mapped to a test case and requirement

14. Test Metrics & Reporting

Metrics collected:

- Test case execution metrics (passed, failed, blocked, not executed)
- Requirement coverage percentage

- Defect density and severity distribution
- Test execution progress
- Defect leakage to later testing stages

Reports:

- Test case execution progress
- Defect leakage
- Defect severity index
- Requirement coverage
- Pass/fail percentage

15. Approvals

Role	Name	Signature / Date
QA Lead	K Sriharsha	
Dev Lead	K Sreekar	
Product Owner	Kirthan G	