TEST PLAN DOCUMENTATION FOR MYSHUTTLE PROJECT

Project: My-Shuttle

Version: 1.0

Prepared by: QE Team

Date: 8/09/2025

1. Test Plan Identifier

MSTP-V1.0-20241213

This document is the master test plan for the MyShuttle project, version 1.0.

2. References

The following documents were used as a basis for this test plan:

- MyShuttle Project Requirements Document v1.2
- MyShuttle System Architecture and Design v1.1
- MyShuttle User Experience (UX) Guidelines v1.0

3. Introduction

MyShuttle is a sample Java application that provides a booking system, admin portal, and a control system for drivers. The application uses entirely open source software, including Linux, Java, Apache, and MySQL, which creates a web front end, an order service, and an integration service.

The objective of this test plan is to define the strategy, scope, and resources required for testing the MyShuttle application. The plan outlines the overall approach to ensure that the application meets the functional, non-functional, and quality standards set by the stakeholders.

4. Test Items

The following components of the MyShuttle application are to be tested under this plan:

- **Web Front End (User Booking Portal):** Version 1.0

- **Admin Portal:** Version 1.0
- **Driver Control System:** Version 1.0
- **Order Service API:** Version 1.0
- **Integration Service:** Version 1.0
- **Database Schema:** Version 1.0 on Azure SQL Database

5. Features to be Tested

This testing effort will focus on the following features from a user's perspective:

- **Functional Testing**:
    - **Login**: Validating core authentication for all user roles.
    - **Dashboard**: Ensuring proper navigation and data display after login.
    - **User Management**: Verifying user creation, updates, deletions, and role-based access control.
- **UI/UX Testing**: Verifying the user interface for consistency, responsiveness, and ease of navigation across different devices and screen sizes.
- **Integration Testing**: Ensuring seamless interaction between Apache Tomcat, Docker, and the database.
- **Performance Testing**: Evaluating the application's response time and stability under normal and peak load conditions.
- **Security Testing**: Ensuring user data is securely handled and protected from vulnerabilities.
- **Compatibility Testing**: Verifying the platform's compatibility across various browsers (Chrome, Firefox, Safari, Edge) and devices (desktop, tablet, mobile).

6. Features Not to be Tested

The following features and areas are explicitly out of scope for this test plan:

- **Underlying Infrastructure Performance**: Testing of the underlying Azure SQL Database performance tuning or the network infrastructure is not included.
- **Third-Party API Load Testing**: While integration with third-party APIs (e.g., payment gateways) will be tested, load testing these external services is not in scope.

- **Live Disaster Recovery**: While deployment rollback procedures will be tested, a full-scale live disaster recovery simulation is not part of this plan.
- **Apache Tomcat Server Hardening**: Security testing will focus on the application layer; server-level security hardening and configuration are assumed to be handled by the infrastructure team.

7. Software Risk Issues

The following are identified as key software risks that could impact the quality of the MyShuttle application. Testing will prioritize these areas.

- **Complex Role-Based Access**: The user management module has complex permissions. There is a high risk of privilege escalation or incorrect access if not tested thoroughly.
- **Data Integrity**: With separate services for orders and integration, there is a risk of data synchronization issues or corruption between the web front end and the backend database.
- **Third-Party Payment Integration**: The system's reliability is dependent on external payment APIs. Any failure or latency in these services could critically impact the booking workflow.
- **Session Management**: Improper handling of user sessions could lead to security vulnerabilities and a poor user experience.

8. Approach

8.1 Test Types

- **Manual Testing**: Exploratory, functional, UI, and integration testing to cover a wide range of test cases manually.
- **Automated Testing**: Automation for repetitive, regression, and smoke test cases to ensure fast and consistent execution.
- **Performance Testing**: Load testing using tools like JMeter to verify the application's behavior under heavy load conditions.

8.2 Testing Levels

- **Unit Testing**: Performed by developers to ensure individual components are working as expected.
- **Integration Testing**: Ensuring smooth integration of different modules like user authentication, dashboard, and database connectivity.

- **System Testing**: Verifying end-to-end functionality of MyShuttle, ensuring all modules work together.
- **User Acceptance Testing (UAT)**: End-users validate the system to ensure it meets business requirements.

8.3 Testing Tools

- **Test Management**: Azure DevOps Test Plans.
- **Automation Tools**: Selenium for UI testing, Postman for API testing, and JMeter for performance testing.
- **Bug Tracking**: Azure DevOps.

9. Item Pass/Fail Criteria

- **Test Case Pass/Fail Criteria**:
    - **Pass**: All steps in the test case are completed, and the actual result matches the expected result.
    - **Fail**: The actual result deviates from the expected result, or the application crashes.
- **Exit Criteria for System Testing**:
    - 100% of critical-severity defects are resolved and verified.
    - 95% of major-severity defects are resolved and verified.
    - 98% of all planned test cases have been executed.
    - 95% of executed test cases have passed.
- **Exit Criteria for UAT**:
    - All UAT test cases pass successfully.
    - The project manager and key stakeholders provide formal sign-off.

10. Suspension Criteria and Resumption Requirements

- **Suspension Criteria**: The entire test execution will be paused if:
    - A critical "showstopper" defect is found that prevents further testing of a major module (e.g., Login functionality is completely broken).
    - The test environment becomes unstable or unavailable for more than 4 hours.

- A new build deployed to the test environment fails more than 50% of the smoke tests.

- **Resumption Requirements**: Testing will resume once:
  - The blocking defect has been fixed, verified, and a new stable build is deployed.
  - The test environment is restored and confirmed to be stable.
  - A new build passes the smoke test suite.

11. Test Deliverables

- **Test Cases**: Comprehensive test cases covering all features of MyShuttle.
- **Test Execution Reports**: Detailed reports showing the pass/fail status of each test case.
- **Defect Logs**: A detailed record of all issues identified during testing.
- **Test Summary Report**: A final report summarizing the testing effort, defect statistics, and overall quality of the application.

12. Test Tasks and Schedule

12.1 Schedule

Phase	Start Date	End Date
Test Planning	2025-8-13	2025-8-16
Interface creation	2025-8-16	2025-8-17
Testcase Creation	2025-8-17	2025-8-17
Test Implementation	2025-8-18	2025-8-18
Verify Testcase	2025-8-19	2025-8-19
Test Completion	2025-8-19	2025-8-19

12.2 Daily Task Plan

Day	Task	Time	Allocated Responsibility
Day 1	Setup test environment, test data preparation	7 hours	QA Engineers, Automation Engineers
Day 2	Functional testing (Login, Dashboard, Fair History)	7 hours	QA Engineers
Day 3	Integration testing (Tomcat, Azure, Database)	7 hours	QA and Automation Engineers
Day 4	Performance and Compatibility testing	7 hours	QA Engineers

12.3 Remaining Test Tasks

This test plan covers all planned testing activities for the v1.0 release of the MyShuttle application. No test tasks are intentionally deferred.

13. Environmental Needs

- **Hardware**:
    - Test workstations with Windows (Windows Server used for backend services).
    - Mobile devices for compatibility testing (iOS and Android).
- **Software**:
    - **Browsers**: Chrome, Firefox, Edge.
    - **Automation Tools**: Selenium, JMeter, Postman.
    - **Test Management**: Azure DevOps.
- **Database**:
    - Azure SQL Database populated with test data.

14. Staffing, Training, and Responsibilities

14.1 Staffing Needs

- **QA Engineers (4)**: Responsible for functional, usability, and manual testing.
- **Automation Engineers (4)**: Tasked with designing and executing automated tests.
- **Project Manager (1)**: Oversees the entire testing process.

14.2 Training Needs

- All team members will require a one-hour walkthrough of the MyShuttle application's latest build before testing begins.
- Automation engineers may require specialized training on new features in Selenium or JMeter if applicable.

14.3 Responsibilities

Activity	Project Manager	QA Engineers	Testers
Test Plan Approval	**A, R**	I	I
Test Case Creation	I	**R, A**	C
Manual Test Execution	I	**R, A**	C
Automated Test Scripting	I	C	**R, A**
Defect Reporting	I	**R**	**R**
Test Summary Report	**R, A**	C	C

Legend: **R -** Responsible, **A -** Accountable, **C -** Consulted, **I -** Informed

15. Planning Risks and Contingencies

Risk	Impact	Mitigation Strategy
Incomplete Test Data	Delays and inaccurate results.	Prepare data in advance, create mock data.
Deployment Failures	Delays and instability.	Test deployment in dev, prepare rollback.
Compatibility Issues	Functionality may break on platforms.	Test early across browsers and devices.

16. Approvals

Role	Name	Signature	Date
Project Manager	Suresh Nanjan		
Automation Engineer	Elangovan		
Automation Engineers	Gayathri		
Automation Engineers	Teja		

17. Glossary

- **UAT**: User Acceptance Testing.
- **Smoke Test**: A preliminary set of tests to reveal simple failures severe enough to reject a prospective software release.
- **Regression Testing**: Retesting previously developed and tested software after a change to ensure that defects have not been introduced.
- **API**: Application Programming Interface.