**Software Test Plan for Hotel B2B (Group 4)**

**Presenter 1: Ludwig Vincent**

1. Introduction

1.1. Purpose

This document outlines the software testing methods and approaches that will be used to ensure the functionality, integration, and performance of the B2B marketplace platform, which connects hotels with local suppliers. This testing will validate the reliability, supplier comparison function, and seamless end-to-end purchasing process.

1.2. Scope

Testing all integral functionalities, including supplier search and comparison, purchase history management, authentication, and contract/payment processes. Non-functional testing will address security, performance, and usability.

1.3. References

* Project proposal.
* API document.
* Hospitality management system - [Mews](https://www.mews.com/)

2. Test Strategy

2.1. Objective

To identify and resolve defects in Hotel B2B to ensure an efficient, secure, and smooth experience for hotel customers and suppliers.

**Presenter 2: Lorenzo Gomez**

2.2. Approach

* Component functionality (Unit testing).
* Service integrations (Integration testing).
* End-to-end workflows (System testing).
* Usability and performance under load (Non-functional testing).

2.3. Tools

* JIRA (Est script creation, defect tracking, and reporting).
* Selenium (Automated functional and regression testing).
* Postman (API testing: authentication, supplier data retrieval, and payments).
* JMeter/LoadRunner (performance testing).
* SonarQube (code quality/security scanning).

2.4. Environments

The software will be tested through dedicated environments with isolated configurations that mimic real-life situations, and features that pass will be deployed to the live production environment.

2.5. Entry and Exit Criteria

Entry:

* Test environment setup is complete.
* Test data (supplier and hotel profiles) is ready.

Exit:

* All test cases are executed.
* All critical defects are resolved.

**Presenter 3: Ian Cressall**

3. Scope of Testing

3.1. Functional Testing

* Unit Testing: Core modules like supplier search, comparison lists, account management, and purchasing flow.
* Integration Testing: Supplier data scraping, payment gateway, and Google login.
* System Testing: End-to-end procurement flow.
* UAT: Hotel managers and suppliers validate usability and business fit.
* Regression Testing: Ensures new features don’t break existing functionality.
* Smoke Testing: High-level verification before each release.
* Security Testing: Authentication (Google login, password reset), supplier data protection, SQL injection, and XSS prevention.

3.2 Non-Functional Testing

* Usability Testing: Ease of supplier filtering and comparison interface clarity.
* Performance/Load Testing: Ensure ≤ 2s response time under 1,500 concurrent users; stable under peak seasonal demand.
* Coding Standards: Validate against project coding guidelines.

4. Consideration of Infrastructure

4.1. Server Configuration

* Tests on a staging environment with 8 GB RAM, 4 vCPUs, scalable to production settings.

**Presenter 4: Tu Ta Aung**

4.2. Database

* Testing on a PostgreSQL/MySQL database with backup and rollback capabilities (including supplier and hotel accounts).

5. Risks or Mitigation Plan

5.1. Risks

* Initial lack of suppliers (thin database for testing).
* Authentication failures with Google login.
* Performance bottlenecks during seasonal spikes.
* Data inconsistency in purchase history.

5.2. Mitigation

* Seed test data with scraped supplier profiles.
* Mock authentication APIs in early stages.
* Run load tests with 2–3× expected peak demand.
* Regular database integrity checks and backups.

**Presenter 5: Chris Paxtor**

6. Resourcing

6.1. Team Composition

* 1 Test Manager.
* 2 Functional Test Engineers.
* 1 Automation Engineer.
* 1 Performance/Security Engineer.

7. Milestones and Deliverables

7.1 Milestones:

* Sprint 1–2: Environment setup, Unit + Integration tests.
* Sprint 3: System testing, Coding Standards + Usability testing.
* Sprint 4: Regression + Smoke testing, Performance baseline.
* Sprint 5: UAT, Load/Stress testing.
* Final Sprint: Fixes, Final Test Report.

7.2. Deliverables

* Test case documentation
* Bug reports and tracking logs.
* Test completion summary.
* Performance & database integrity reports.