**Test plan for Kushi travels online cab booking application**

**Introduction**

This document outlines the test plan for the Kushi Travels website, which provides services like flight booking, hotel reservations, and holiday packages. The goal of this testing effort is to ensure that all functionalities are working as expected and that the website delivers a smooth and secure user experience.

**1. Objectives**

* To verify that the Kushi Travels website meets all specified functional and non-functional requirements.
* To identify and address any defects in the system before deployment.
* To ensure the website's usability, performance, and security are up to industry standards.

**2. Scope**

* **Register**
* **Login**
* **Account**
* **Confirmation**
* **Categories**
* **Logout**
* **Links**
* **Vehicle type**

**3.Test Approach**

* **Functional Testing**: Testing the core features such as user registration, search, booking, and payment processing.
* **Usability Testing**: Evaluating the user interface, navigation, and accessibility.
* **Performance Testing:** Load testing, stress testing, and evaluating response times under different conditions.
* **Security Testing:** Ensuring secure handling of user data, transaction security, and protection against vulnerabilities.
* **Compatibility Testing:** Testing the website across different browsers, devices, and operating

**4.Test Environment:**

**Hardware Requirements**:

Devices: Android and iOS smartphones, tablets, laptops, and desktops.

**Software Requirements**:

Operating Systems: Windows, macOS, Linux, iOS, and Android.

Browsers: Chrome, Firefox, Safari, Edge, and Opera.

**Testing Tools**: Test management tools (e.g., JIRA, TestRail), automation tools (e.g., Selenium, Appium), performance testing tools (e.g., JMeter), and security testing tools (e.g., OWASP ZAP).

**5.** **Entry and Exit Criteria:**

1. **Requirements Documentation:**

All functional and non-functional requirements are clearly documented and signed off by stakeholders.

1. **Test Environment Setup:**

The test environment, including necessary hardware, software, network configurations, and tools, is fully set up and verified to be working as expected.

1. **Test Data Prepared:**

Relevant test data (e.g., user accounts, restaurant details, menu items) is created and validated for use in testing.

1. **Test Cases Developed:**

All test cases are written, reviewed, and approved. These test cases should cover all the features in scope, including functional, usability, performance, and compatibility tests.

1. **Test Team Readiness:**

The testing team is fully trained, briefed on the project scope, and ready to execute test cases.

1. **Dependencies Resolved:**

All dependencies, such as third-party integrations (excluding the payment gateway), API availability, and inter-module dependencies, are resolved.

**Exit Criteria:**

1. **Test Execution Completion:**

All planned test cases have been executed, including functional, usability, performance, compatibility, and regression tests.

1. **Defect Resolution:**

All critical and major defects are identified, resolved, and verified. There should be no unresolved high-severity defects. Low-priority defects are documented for future releases.

1. **Test Coverage Achieved:**

The required test coverage is met, with all critical features thoroughly tested and validated.

1. **Performance Benchmarks Met:**

The application meets the required performance benchmarks, including response time, load handling, and stability under different network conditions.

1. **Final Test Report:**

A final test report summarizing the test execution, defects found and resolved, and overall quality assessment is prepared and reviewed by stakeholders.

1. **User Acceptance Testing (UAT):**

UAT is completed successfully, with sign-off from business stakeholders, indicating that the application meets business requirements and user expectations.

1. **Approval from Stakeholders:**

Final approval is received from key stakeholders, including the QA Lead, Project Manager, and Product Owner, indicating that the application is ready for release.

**6. Test Schedule**

| **Phase** | **Start Date** | **End Date** | **Responsible** |
| --- | --- | --- | --- |
| Test Planning | Sept 11, 2024 | Sept 13, 2024 | Test Lead, Project Manager |
| Test Design | Sept 14, 2024 | Sept 18, 2024 | Test Analysts, QA Engineers |
| Test Environment Setup | Sept 19, 2024 | Sept 20, 2024 | DevOps, QA Engineers |
| Test Execution | Sept 21, 2024 | Oct 5, 2024 | QA Engineers |
| Test Closure | Oct 6, 2024 | Oct 7, 2024 | Test Lead, QA Engineers |

**7. Test Deliverables**

* Test Plan Document
* Test Cases and Test Scripts
* Test Data
* Defect Logs
* Test Summary Report
* Final Test Report
* UAT Sign-off

**8.Roles and Responsibilities**

**Test Manager**: Oversee the entire testing process, manage resources, and ensure timelines are met.

* Communicate with stakeholders and provide updates on testing progress.

**Test Lead:** Manage daily testing activities, assign tasks, and monitor progress.

* Review test cases and reports for quality and completeness.

**Test Engineers**: Design and execute test cases.

**9.Risks and Mitigation**

Mitigation: Maintain regular communication with the development team and stakeholders, and continuously review and update requirements.

Test Environment Issues: Problems with the test environment could delay testing activities.

Mitigation: Set up the test environment early and perform regular checks to ensure stability.

Resource Constraints: Limited testing resources may cause delays.

Mitigation: Prioritize testing tasks and engage cross-functional teams to distribute the workload.

**10.Approvals:**

1.MoinKhan (Test Manager) **–** Reviews and approves the overall test plan, including development of test cases, and ensure timelines are met. Communicate with stakeholders and provide updates on testing progress

2.Thanushree (Test Engineer) – Creating the test plan, , including development of test cases and executing test cases.