# Test Plan

1. Objective

2. Scope

3. Testing Strategy

4. Unit Testing

5. Integration Testing

6. Performance and Stress Testing

7. UAT

8. Automation and Regression Testing

**1. Objective**

The objective of the Test plan document is to identify the various functions of the application, and making sure that the users are able to access the various options available in the page without encountering issues.

**2. Scope**

The following functions are under scope of the application

|  |  |
| --- | --- |
| S.No | Function |
| 1. | Select the location |
| 2. | Select dates |
| 3. | Rent a car |
| 4. | Access the perks / promotions |
| 5. | Link to explore rentals in various cities |
| 6. | Navigate to Business travel page |
| 7. | Link to business travel |
| 8. | Find a Rental button |
| 9. | Support and other links at the bottom of the page |
| 10. | Formatting of the different sections of the page |

**3. Testing Strategy**

Three environments are to be created to develop and test the application. The Dev environment to be used by the developers for creation and unit testing, QA environment to contain subset of data from production and Production environment that would be used by actual customers. The following applications are to be used in various parts of testing. The various scenarios that are to be performance tested and automation tested are to be covered.

1. Jira – for creating tickets to document all the features and adding requirements and assigned it to designated team members

2. Postman – for sending API calls while testing

3. MySQL- for maintaining the tables and accessing them. Also create separate access for dev, QA and Prod

4. Jmeter – for performance testing and load testing of application

5. Selenium for automation of regression test cases

**Risk and Issues**

1. Access and Availablity of the QA and Dev environment

* Mitigation -Ensure access and availability before the start of timeline of each phase
* 2. Resource Availability – mitigation : plan with 10% buffer time

5. Unit testing and Integration Testing

Each component of the software application would be tested by the developer. Once the end to end functionality have been developed it would be sent to the QA for testing. And the corresponding ticket will be moved to QA.

Once the application functionality is available in QA the tester would be able to test the functionality and reopen the tickets if they do not match the requirements as per the tickets in JIRA.

Once the functionalities are working are per the requirement, the tester can work on creating the performance testing scripts and automation scripts.

Performance Testing Scenarios

Identify the duration and the peak load details

|  |  |
| --- | --- |
| S.no | Scenarios |
| 1. | Rent a car with highly visited locations and off season duration |
| 2. | Rent a car with highly visited locations and peak season duration |
| 3. | Rent a car with less frequently visited locations and peak season duration |
| 4. | Rent a car with less frequently visited locations and off - peak duration |
| 5. | Login and Logout |
| 6. | Access Booking |
| 7. | Make changes to the booking |
| 8. | Cancel booking |
|  |  |

Automation Testing

Develop scripts to test the below

|  |  |
| --- | --- |
| S.no | Scenarios |
| 1. | Rent a car with highly visited locations and off season duration |
| 2. | Rent a car with highly visited locations and peak season duration |
| 3. | Rent a car with less frequently visited locations and peak season duration |
| 4. | Rent a car with less frequently visited locations and off - peak duration |
| 5. | Login and Logout |
| 6. | Access Booking |
| 7. | Make changes to the booking |
| 8. | Cancel booking |

**Critical scenarios**

1. Rent a car with highly visited locations and peak season duration

1. The user is able to book the car – positive scenario

2. Test steps

1. Enter the location

2. Enter the dates

3. Click on rent

4. Select the type of car

5. Confirm the booking by entering the personal details and payment details

2. Rent a car with highly visited locations and peak season duration

1. The user is unable to book the car – negative scenario

2. Test Steps

1. Enter the location

2. Select the dates

3. Click on rent

4. Display no availability during the time

5. Display the option for the user to modify the dates