**Test Plan Document**

**for**

**‘Rent a Car’ for Lyft**

1. **Product Introduction & Overview:**

**Introduction:**

The purpose of this test plan is to ensure the reliability, functionality, and usability of the "Rent a Car" functionality on the Lyft website (<https://www.lyft.com/rider/rentals>).

**Scope**: This test plan covers the testing of the "Rent a Car" feature, including the login/register a new user , booking, payment, and cancellation processes and also view order details.

**Test Objectives**:

Verify that users can successfully load the website and logged into to the website(if registered user) or register into the site if new user.

Verify that users can successfully reserve and rent a car.

Ensure that payment processing is secure and reliable.

Confirm that users can cancel reservations without issues.

Validate the usability and user experience of the "Rent a Car" feature.

**Test Data**: Test data will include various scenarios of car rental, payment methods, user profiles, and geographical locations.

**Test Strategy:**

The objective of the test is to verify that the functionality of ‘Rent a Car’ feature of Lyft website works according to the specifications.

Types of testing needs to perform:

1. Unit Testing (To be done by Developer own)
2. Exploratory Testing (to make sure critical defects are removed before the next levels of testing can start.
3. Functional Testing (To be checked functionality of each functional points.
4. System Testing (To be checked end to end testing of the product.
5. Retesting and Regression testing (To be checked after developer fix bug/issues whether or not other points to be breaking.
6. Automated testing will cover functional and regression testing.

**Testable Items (Features to be tested): Or Test scenario: (Valid logged in user)**

1. **Reservation and Booking:**

1.Verify that users can search for available rental cars.

2. Ensure users can select a car and view its details.

3. Confirm that users can choose rental dates and times.

4. Validate that users can successfully book a selected car.

5.Check for confirmation messages and emails after booking.

1. **Payment Processing**
2. Test different payment methods (credit card, PayPal, etc.) for successful transactions.
3. Ensure that payment processing is secure and compliant with industry standards.
4. Validate the handling of payment failures and error messages.
5. **Reservation Management**
6. Verify users can view and manage their existing reservations. Confirm that users can cancel reservations without issues.
7. Check for appropriate refund processes if applicable.
8. **Usability and User Experience**
9. Evaluate the overall user interface for intuitiveness and responsiveness.
10. Verify that the website is accessible and usable on both desktop and mobile devices. Test the loading times of pages and features.

**Test Execution and Defect tracking:**

Tester or QA team can execute all the testable items or test cases based on test cases document and log the defect when the test cases fails in a Bug tracking tool “Service Desk”.

Issues priority / severity can be categorized as Critical (0), High (1), Medium (2) and Low (3 may be cosmetic).

After developer fix the issues, they are mark that ‘Resolved’ and then QA/tester can again perform retest or regression testing to be closed those issues.

**Test Environment:** The test environment will include various web browsers (Chrome, Firefox, Safari) and mobile devices (iOS and Android).

**Test completion :**  Once most of the high/critical and medium priority issues are closed then we can deploy or make that product to be Go-live.