**Test Strategy – Identify Car Wash Services**

**JustDial**

**Identify Car Wash Services**

**Test Strategy**

**Author:** Auto Mates

**Date:**

**Version Number:** 001

**Test Strategy – Web Development Courses**

**Table of Contents**

1. **INTRODUCTION**  3
   1. Purpose 3
   2. Target Audience 3
2. **SYSTEM DESCRIPTION**  3
3. **TEST DELIVERABLES** 4
4. **TEST TYPES** 4
5. **MULTIBROWSER TESTING** 4
6. **TEST DATA** 4
7. **RESOURCES REQUIRED** 5

7.1 Hardware 5

7.2 Software 5

**Test Strategy – Web Development Courses**

**1.0 Introduction**

**1.1 Purpose**

The Test Strategy document outlines the types and levels of testing to be performed, as well as the methodology that will be followed during different testing phases. It describes how the testing process will ensure that all key features—both new and existing—of the system are validated at the appropriate level.

Additionally, this document sets the boundaries for testing activities, helping in the creation of test conditions, test cases, automation efforts, and time estimates.

It also acts as a means of communication to help identify and address testing-related issues as early as possible within the project lifecycle.

**1.2 Target Audience**

Business Unit (BU) SME

Academy Coach

Technical Trainer

**2.0 System Description**

Justdial.com is a comprehensive local search engine in India that allows users to discover businesses, services, and professionals across a wide range of categories including healthcare, automotive, hospitality, education, and more. The platform supports advanced features like location-based search, filtering by ratings and reviews, and interactive UI elements such as dropdowns, sliders, and dynamic menus. It also includes service booking options, free listing registration, and category-specific submenus, making it a rich candidate for functional UI automation using tools like Selenium.

However, automation of Justdial.com comes with certain limitations. Key workflows such as user login or signup require mobile number verification via OTP, which cannot be automated due to security constraints. Additionally, dynamic content loading, third-party widgets, and real-time data integrations may pose challenges for stable automation. Despite these, many functional areas—like search workflows, form validations, and menu discovery—can be effectively automated using a structured framework with Selenium, TestNG, Apache POI, and Cucumber for data-driven testing and reporting.

**3.0 Test Deliverables**

|  |  |
| --- | --- |
| Project Phase | Deliverables |
| Test Planning | **·** Test Strategy Document |
| Test Analysis & Design | **·** Test Conditions  **·** Test Cases  **·** Test Data  **·** Test Environment  **·** Automated Test Scripts |
| Test Execution | **·** Test Logs |
| Test Completion | **·** Test Summary Report |

**4.0 Test Types**

The following are the types of Testing used in the Web Development courses

1. Functional Testing
2. Manual Testing
3. Automation Testing
4. Smoke Testing

**5.0 Multi browser Testing**

Based on the requirements for the application the testing is done on two browsers to ensure the working of the application in multiple browsers. The two browsers used for testing are Chrome and Edge.

**6.0 Test Data**

Test data for the application is provided from excel file. The test data are stored in the excel file, XML and JSON and provided during run time. Data driven library creation is done using Apache POI and property file is used to keep the URL, Browser, Username and Password.

**7.0 Resources Required**

**7.1 Hardware**

* PC

**7.2 Software**

* Windows 7 and above
* Chrome 60 and above
* MS Excel
* Selenium Web drivers
* Selenium Grid
* BDD Cucumber
* Jenkins
* Apache POI
* TestNG