To create test strategy, test plan and test cases for one of the modules in URBANLADDER.com

URBANLADDER

**Test Strategy**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**TABLE OF CONTENTS**

1. **Scope……………………………………………………………..**
2. **Test Approach…………………………………………………...**
3. **Test Environment……………………………………………….**
4. **Testing Tools…………………………………………………….**
5. **Release Control………………………………………………….**
6. **Risk Analysis…………………………………………………….**
7. **Review and Approvals…………………………………………..**
8. **SCOPE**

The release of software is being released to bring the URBANLADDER application on to the ABC. Testing will cover the functional testing of the application. Functionality for this release is detailed in the URBANLADDER applications Requirements specifications documents. Installation will be tested on the different platforms as described in the Requirements Specification. The testing for this will cover the installation on these platforms, as well as a set of critical functions to determine that the code will work on all platforms. Document will be reviewed by ABC. Document will be approved by ABC. Testing of the module deals of the week should be completed by December 1st 2021.

1. **TESTING APPROACH**

The testing approach for this release shall be done in a fashion that will accommodate the current functionality in products being developed for URBANLADDER. Testing will cover functionality testing for the applications changes through the use of the test interface. This will validate base functions of the new code as it relates to the standard model of presentation for data and user entered data. Unit testing is testing performed to determine that individual program modules perform per the design specifications and to check whether the module deals of the week satisfies the requirements. Testing will be starting at November 29 2021. Testing will be done by ABC using the tool Jasmine.

1. **TEST ENVIRONMENT**

The Test Plan should enumerate environments that will be used to test an application. A test environment description should specify properties that facilitate:

* Chrome browser,
* Chrome driver executable file,
* Ide for Jasmine,
* node,
* node packages of Jasmine.
* Microsoft Windows 7 with Service Pack 1

1. **TESTING TOOL**

**JASMINE**

Jasmine is an open-source JavaScript framework, capable of testing any kind of JavaScript application. Jasmine follows Behaviour Driven Development (BDD) procedure to ensure that each line of JavaScript statement is properly unit tested. By following BDD procedure, Jasmine provides a small syntax to test the smallest unit of the entire application instead of testing it as a whole.

Jasmine does not depend on any other JavaScript framework. Jasmine does not require any DOM. All the syntax used in Jasmine framework is clean and obvious. Jasmine is heavily influenced by Rspec, JS Spec, and Jspec. Jasmine is an open-source framework and easily available in different versions like stand-alone, ruby gem, Node.js, etc.

1. **RELEASE CONTROL**

Test execution for all modifications in that release will be ensured by a release management plan with suitable version history. As a result, establish a build management process to determine where the new build should be available when it will be delivered, who will deploy it, how to stop the release in the event of errors, and so on.

1. **RISK ANALYSIS**

Make an effort to anticipate all potential hazards associated with your project. Make a clear plan to avoid such risks, as well as a contingency plan in case they occur.

1. **REVIEW AND APPROVALS**

Managers, technical team leaders, business development, and software development teams must all examine and finalize strategy. An overview of review modifications, as well as an authorized date, name, and comment, should be put at the top of the document. In addition, if significant changes occur throughout the testing process, the Test Strategy document can be revised.

URBANLADDER

**Test Plan**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version No. | Change Date | Author | Description |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**TABLE OF CONTENTS**

1. **Introduction**
   1. Scope…………………………………………………………………………………
   2. Objective……………………………………………………………………………
   3. Roles and Responsibilities…………………………………………………
2. **Test Methodology**
   1. Overview………………………………………………………………………..
   2. Testing method……………………………………………………………..
   3. Test Completeness…………………………………………………........
3. **Test Deliverables……………………………………………….**
4. **Resource and Environment Needs**
   1. Testing Tools………………………………………………………………………
   2. Test Environment………………………………………………………………
5. **INTRODUCTION**
   1. **Scope**

The release of software is being released to bring the URBANLADDER application on to the ABC. Testing will cover all the functional testing of the mentioned application. Functionality for this release is detailed in the URBANLADDER applications Requirements specifications documents. Installation will be tested on the different platforms and the testing for this will cover the installation on the platforms, as well as a set of critical functions in order to determine the working of the code on all platforms. Documents will be reviewed by ABC. Document will be approved by ABC. Testing of the modules is to be completed by December 1st 2021.

* 1. **Objective**

To ensure the testing for the URBANLADDER application to meet all the specifications provided by the client.

* 1. **Roles & Responsibilities**
* The team members have to ensure that the testing for the URBANLADDER application has to meet all the criteria as provided by the client.
* The bugs or the issues are to be identified and has to be fixed before go live.

1. **TESTING METHODOLOGY**
   1. **Overview**

The test plan covers an environment that is used to test the URBANLADDER application.

The test methodology for this project includes:

* Chrome browser,
* Chrome driver executable file,
* Ide for Jasmine,
* node,
* node packages of Jasmine,
* Microsoft Windows 7 with Service Pack
  1. **Testing Methodology**

Testing will cover the functionality testing for the applications changes through the use of the test interface which helps in validating the base functions of the new code. Thereafter the unit testing is performed which is done to determine whether the individual programs perform and satisfies all the design requirements. Testing will be started on November 29th 2021 using the tool Jasmine.

* 1. **Test Completeness**

The criteria that complete the testing of the project are as follows:

* 100% test coverage
* All test cases being executed
* All bugs and issues are fixed

1. **TEST DELIVERABLES**

The test arifacts that will be delivered during the different phases of the testing lifecycle are as follows:

* Test Strategy
* Test Plan
* Test Cases
* Test Scenarios
* Test results
* Defect reports

1. **RESOURCES AND ENVIRONMENT NEEDS**
   1. **Testing Tools**

Jasmine is the main testing tool used here. Jasmine is an open-source JavaScript framework, capable of testing any kind of JavaScript application. Jasmine follows Behaviour Driven Development (BDD) procedure to ensure that each line of JavaScript statement is properly unit tested. By following BDD procedure, Jasmine provides a small syntax to test the smallest unit of the entire application instead of testing it as a

whole.

* 1. **Test Environment**

The Test Plan should enumerate environments that will be used to test an application. A test environment description should specify properties that facilitates the following:

* Chrome browser,
* Chrome driver executable file,
* Ide for Jasmine,
* node,
* node packages of Jasmine,
* Microsoft Windows 7 with Service Pack