

**EEC 521 / CIS 634 – Software Engineering**

**Course Project**

**Submitted by,**

**Ruthvi Metical, 2868919**

**Sreekanth Mullamuri, 2872139**

**Poojith yerra Anand, 2833835**

***TEST SPECIFICATION***

**1.0 Introduction**

The Test Specification provides a comprehensive overview of the testing process for the Unitravells software. This document encompasses both the test plan and the test procedure, ensuring a systematic approach to validating the software's functionality.

**1.1 Goals and objectives**

The primary goal of the testing process is to ensure the quality, reliability, and security of the Unitravells software. The key objectives include:

1. Verify the correctness of implemented features and functionalities.
2. Identify and address any defects or issues in the software.
3. Validate the software against specified requirements.
4. Ensure optimal performance and responsiveness.

**1.2 Statement of scope**

* **Functional Testing:** Validating each functional requirement against expected behaviour.
* **Performance Testing:** Evaluating the system's responsiveness and efficiency.
* **Security Testing:** Verifying the robustness of security measures implemented.
* **User Interface Testing:** Ensuring the user interface meets design standards and usability expectations.
* **Unit Testing:** Unit testing emphasizes the verification effort on the smallest unit if software design i.e., a software component or module. It is a dynamic method for verification, where program is actually compiled and executed. Unit testing is performed in parallel with the coding phase.
* **Integration Testing:** In integration testing a system consisting of different modules are tested for problems arising from component interaction.
* **Validation Testing:** It provides final assurances that software meets all functional, behavioral, performance requirements.

**Exclusions:**

Certain functionalities or behaviours may not be explicitly tested, including:

* **Third-Party Integrations:** Testing of external systems beyond the project's control.
* **External Hardware Compatibility:** Validation on hardware configurations not within the specified environment.

**2.0 Test Plan**

This section outlines the overall testing strategy and project management issues essential for effective test execution.

**2.1 Software to be tested**

**Software Identification:**

The software to be tested is Unitravells. Exclusions include third-party integrations and external hardware components.

**2.3 Testing tools and environment**

**Test Environment Description:**

The test environment for Unitravells includes:

* **Testing Tools:** Selenium for automated testing, Jira for issue tracking.
* **Simulators:** Mobile device simulators for testing mobile responsiveness.
* **Specialized Hardware:** None required for software testing.
* **Test Files:** Sample test data files for various test scenarios.

**2.4 Test schedule**

**Detailed Testing Schedule:**

The testing schedule is structured as follows:

* **Unit Testing:** [11/16] to [11/19]
* **Integration Testing:** [11/20] to [11/23]
* **System Testing:** [11/24] to [11/27]
* **User Acceptance Testing (UAT):** [11/30] to [12/05]

**3.0 Test Cases**

This section presents a complete list of test cases for the Unitravells software. Each test case is documented using the following template:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **TEST INPUT** | **EXPECTED OUTPUT** | **DESCRIPTION** |
| 1 | Search for Trip packages | List of available trip/tour packages | Check that users can search for and view a list of available trip/tour packages per person |
| 2 | Search for accommodation | List of available accommodations | Check that users can search for and view a list of available accommodations. |
| 3 | User Details | Message to admin for contacting and about trip details | User fill contact or trip detail form for package enquiry or any other reason |
| 4 | Admin Access | Admin can access entire website workflow | Admin can add trip/tour packages, gather user contact details and contact them back regarding their message. |

These test cases cover a range of scenarios to thoroughly assess the software's functionality and performance.

This section enumerates a complete list of test cases for the software. A template for test cases is as follows.

**Test cases for User Login:**

|  |  |
| --- | --- |
| **ID** | **Login\_1** |
| Test Input | Valid username, invalid password |
| Expected Output | Error message saying invalid credentials |
| Description | User has to enter his/her credentials to login into the application. If these credentials are valid, he/she will be logged in. Else, error message should be displayed. |

|  |  |
| --- | --- |
| **ID** | **Login\_2** |
| Test Input | Invalid username, valid password |
| Expected Output | Error message saying invalid credentials |
| Description | User has to enter his/her credentials to login into the application. If these credentials are valid, he/she will be logged in. Else, error message should be displayed. |

|  |  |
| --- | --- |
| **ID** | **Login\_3** |
| Test Input | Invalid username, invalid password |
| Expected Output | Error message saying invalid credentials |
| Description | User has to enter his/her credentials to login into the application. If these credentials are valid, he/she will be logged in. Else, error message should be displayed. |

|  |  |
| --- | --- |
| **ID** | **Login\_4** |
| Test Input | Valid username, valid password |
| Expected Output | User should be able to log into the application |
| Description | User has to enter his/her credentials to login into the application. If these credentials are valid, he/ she will be logged in. Else, error message should be displayed. |

**Test cases for User Registration:**

|  |  |
| --- | --- |
| ID | Register\_1 |
| Test Input | Valid details like FirstName, LastName, Address, City, ZIP Code, Country, Password, Confirm Password, Mobile Number, Email ID. |
| Expected Output | User should be able to register into the application |
| Description | User has to enter his/her valid details to register into the application. |

|  |  |
| --- | --- |
| ID | Register\_2 |
| Test Input | FirstName, LastName, Address, City, ZIP Code, Country, Password, Confirm Password, Mobile Number, Email ID. |
| Expected Output | Error message is shown to fill all the fields. |
| Description | If user does not enter any one of required fields, error message is shown to fill all the details. |

**Test cases for Payments unsuccessful:**

|  |  |
| --- | --- |
| ID | Payments\_1 |
| Test Input | Invalid Card Number or CVV, Expiry Date |
| Expected Output | Payment is not successful. |
| Description | Payment failed please try again |

**Test cases for Payments successful:**

|  |  |
| --- | --- |
| ID | Payments\_1 |
| Test Input | Card Number, CVV, Expiry Date |
| Expected Output | Payment is successful. |
| Description | Payment has to go through successfully while placing the order. |

**Test cases for successful destination search:**

|  |  |
| --- | --- |
| ID | Search\_id1 |
| Test Input | Enter the name of destination |
| Expected Output | search successful and page is redirected to it |
| Description | The page is redirected to new page where you can perform necessary tasks |

**Test cases for unsuccessful destination search:**

|  |  |
| --- | --- |
| ID | Search\_id2 |
| Test Input | Enter the name of destination (if entered destination name is not correct) |
| Expected Output | search is not successful, please enter valid spot |
| Description | Error message is displayes |

**Test cases for successful Accommodation search:**

|  |  |
| --- | --- |
| ID | Ac1 |
| Test Input | Test applying filters (e.g., price range, amenities, ratings) and check if the results match the filter criteria.  Test sorting options (e.g., price low to high, high to low, ratings) and confirm the correct order of results. |
| Expected Output | search is successful |
| Description | Available options are displayed |