**TEST CASES AND TEST PLANS**

**PROJECT NAME: REAL ESTATE APPLICATION.**

**TEAM MEMBERS:**

* **VARNITHA REDDY**
* **SHALINI**
* **SHELLY CHOUDARY**
* **SRAVANI ATLURI**
* **GRACE MANDADA**

**REVISION HISTORY OF TEST PLANS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version**  **No.** | **Description of Releases and modifications** | **Prepared by** | **Approved by** | **Date of approval** |
| **01** | **Initial release** | **Varnitha** | **Shelly** | **04/13/2020** |
|  |  |  |  |  |

**TABLE OF CONTENTS**

1. Introduction……………………………………………………………………..........................03
2. References……………………………………………………………………………………………….03
3. Test plans…………………………………………………………………………………………………04
4. Test cases…………………………………………………………………………………………………06

**Introduction:** Real Estate Application will help It’s users to look at the available properties where the properties are separated into different types like apartments, condominiums and houses. And the admin can post the properties with the price, and the features of the specific property. Where the features are different for different type of properties.

**Reference Document:**

* Software requirements document.
* Software quality assurance plan document.
* Software development plan document.

**Software Test Environment:**

* NetBeans IDE for developing.
* Java version 12 for coding.
* Scene builder for designing.
* Java FX for GUI.
* SQL for Database.

**Objective of Software Testing:** The objective of this software testing is:

* To provide defect free software as possible
* To provide a quality software to the customer.
* To check whether we are getting the expected results.
* To ensure it satisfies system requirement specifications.

**Test Case Preparation:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Person Responsible for preparing the Tests** | **Probable reviewer** | **Schedule of preparation**  **(specify the dates)** |
| **Unit test for software units** | Program author:  Shelly | Shalini | 03/13/2020 |
| **Integration tests for each module** | Project leader:  Shalini | Shalini | 03/16/2020 |
| **System tests** | Software designer: Varnitha | Sravani | 04/02/2020 |
| **Acceptance tests** | Software project manager: Grace | Sravani | 04/10/2020 |

**Tests to be conducted for the project:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **Probable testers** | **Test objectives** | **How to test** | **Criteria for completion** |
| **Unit testing** | Sravani | Make sure that the code is working properly without defects and is in acceptable quality. | Testing by taking piece of code and testing it in ide by creating example project (white box testing) | 85% code has been tested. |
| **Integrated testing** | Shalini | Make sure that the interfaces are working properly, and application navigates from one page to other without any problem. | Using black box testing and test cases | 90% test cases are pass. |
| **System testing** | Shalini  Grace | To evaluate whether we are getting the expected output and ensures it is meeting specific requirements. | Using system testing test cases And black box testing. | 80%test cases are pass. |
| **Functional testing** | Shalini | Functional testing ensures whether application is giving the expected output for the given input. | By giving example inputs and verifying the outputs. | 97% of the test cases are pass. |

**Regression testing strategy:** In this application regression test only takes place only when a new functionality is added to check if existing functionalities are not affected. So following steps must take place

* Re-execute the previously performed tests.
* Compare the current results with the previous results.

**Escalation mechanisms:** Escalation takes place when an issue is unresolved for a longer time. User unable to check the property prices and upload property, which is a serious issue to continue. The issue is raised by the user and an escalation is raised to the quality department like the backend team responsible for the issue. This issue is raised to high priority through email. Once the backend team check the issue, they get back to the users with an email with time resolution.

**Progress strategies:** Progress explains how far you’ve gone through completion of the project. Our ongoing project is estimated to complete by 5th May. The current status of the project is the team members are testing the use cases. Every week one of us check the documents and the other four writes the document.

**Risk Identified:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk id** | **Risk Description** | **Probability of occurrence** | **Mitigation plan** |
| Id1 | Weak authentication | low | User should be prompted with more information |
| Id2 | Insecure direct object references | High | Unauthorized user must not have access |
| Id3 | Improper navigation | low | User must use exact location mentioned |
| Id4 | Loosing data | high | Update the data as backup |

**Tools to aid in testing:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of the tool** | **Purpose** | **Administrator** | **Reference to tool documentation** |
| **Smartsheet** | Defect reporting, resolution and defect metrics | grace | Available in the tool itself |
| **NetBeans for programming languages** | Unit testing | shelly | Available in the IDE itself |
| **SQL server** | Load testing | Varnitha | Available on the internet |
| **Microsoft office** | Prepare test cases, test logs and reports and carry out analyses | grace | Available in the app itself |

**Proposed analysis for the project:**

|  |  |  |
| --- | --- | --- |
| **Analysis** | **Person responsible for carrying out the analysis** | **Schedule for carrying out the analysis** |
| **Defect injection rate for programmers** | Sravani, Varnitha | Once every week on a Thursday  last working day: 04/02/2020 |
| **Rework effect for defect resolution** | Sravani, Varnitha | Once every week on a Monday  Last working day: 03/30/2020 |
| **Defect category analysis** | Sravani, Varnitha | Once every week on a Wednesday  Last working day: 04/01/2020 |
| **Defect by origin analysis** | Grace | Once every week on a Tuesday  Last working day:  03/30/2020 |

**LIST OF TEST CASES**

**Nonfunctional requirement test cases:**

**TESTCASE: 1**

**Project id: Real Estate Application**

**Module name: Non-Functional Requirements Validation**

**Component to be tested: Complete Application.**

**Type of component: screens**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **TST#1** | Verify performance. | Application must be able to navigate with in second from one page to other page. | Application can navigate with in second from one page to other page. | pass |
| **TST#2** | Verify security. | No third party must be able to access the password of the user or admin.  Only admin and registered user must be able to login to the application.  No user can register with the already existed username.  If the user entered the password incorrect for three times his account must get blocked. | No third party can access the password of the user or admin.  Only admin and  And user can login, user cannot register with the same username, user must enter correct password. | pass |
| **TST#3** | Verify reliability. | Application must be able to work under all conditions. | Application can run under all the conditions. | pass |
| **TST#4** | Verify Reusability. | Developer must be able to reuse the code anywhere. | Developer can reuse the code anywhere. | pass |
| **Tst#5** | Verify modifiability. | Developer must be able to add new features to the application. | Developer can add new features to the application. | pass |
| **Tst#6** | Verify portability. | Application must be able to run in any platform. | Application can run in any platform.  Eg: IOS, windows etc., | pass |

**TESTCASE: 1**

**Project id: Real Estate Application**

**Module name: home page validation**

**Component to be tested: home page**

**Type of component: screen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **TST#1** | Verify home page has admin login, user login and user registration and login button | User, admin and user registration tabs should be present. | User, admin and user registration tabs should be present. | pass |
| **TST#2** | Click on the registration tab and verify whether it is navigating to the registration page which has registration form. | Registration form should be displayed. | Registration form should be displayed. | pass |
| **TST#3** | Click on the admin tab and verify whether it is going to admin login page. | Admin login page should be displayed. | Admin login page should be displayed. | pass |
| **TST#4** | Click on the user tab and verify whether it is navigating to user login page. | User login place should be displayed. | User login place should be displayed. | pass |

**Testcase: 02**

**Project id: Real Estate Application.**

**Module name: Admin Login**

**Component to be tested: Home Page**

**Type of component: Screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **TST#1** | Verify username and password text fields were present in admin login page. | Username and password fields should be present | Username and password fields are present. | pass |
| **TST#2** | Enter username and password of the admin | Admin’s username and password should be entered. | Admin’s username and password are entered. | pass |
| **TST#3** | Click on Login button. | Login success message should be displayed, and screen should be navigated to admin page. | Login success message is displayed, and screen is navigated to admin page. | pass |
| **TST#4** | Enter invalid password. | If invalid passwords entered three admin account should be locked | Admin account is locked after entering invalid password for three times | pass |

**Test Case: 03**

**Project id: Real Estate Application.**

**Module name: User Login**

**Component to be tested: Home Page**

**Type of component: Screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **TST#1** | Verify username and password text fields were present in user login page. | Username and password fields should be present | Username and password fields are present. | pass |
| **TST#2** | Enter username and password of the user | User’s username and password should be entered. | user’s username and password are entered. | pass |
| **TST#3** | Click on Login button. | Login success message should be displayed, and screen should be navigated to user page. | Login success message is displayed, and screen is navigated to user page. | pass |
| **TST#4** | Enter invalid password. | If invalid passwords entered three admin account should be locked | Admin account is locked after entering invalid password for three times | pass |

**Test case: 04**

**Project id: Real Estate Application**

**Module name: user registration**

**Component to be tested: user registration page and home page**

**Type of component: screen**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify the registration button in the home page. | Register button should be present on home screen | Register button is present on home screen | pass |
| **Tst#2** | Click on the registration button. | User should be able to click the button and the screen navigates to registration form screen. | Button is clicked and registration form page is displayed | pass |
| **Tst#3** | Verify all the fields In the registration form. | All the fields should be editable by the user | Fields are editable by the user. | pass |
| **Tat#4** | Click on the submit button. | Submit button should be clicked and registration success popup should be displayed. | Submit button is clicked. And it will show registration successful popup | pass |

**Test case: 05**

**Project id: Real Estate Application**

**Module name: Admin home page.**

**Component to be tested: Admin home page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify three buttons create property, view/edit property and logout | Admin must be able to view three buttons create property, view/edit property and logout | Admin can see the three buttons create property, view/edit property and logout in the admin home page. | pass |
| **Tst#2** | Click create property button. | Admin must be able to navigate to the create property menu after pressing the create property button. | Admin can view the create property menu page | pass |
| **Tst#3** | Click view/edit property button. | Admin must be able to view the view/edit properties menu page after pressing view/edit property button. | Admin can view the view/edit property menu page. | pass |
| **Tst#4** | Click logout button. | Admin must be able to logout from the application after pressing the logout button. | Admin can logout from the application. | pass |

**Test case: 06**

**Project id: Real Estate Application**

**Module name: create property menu.**

**Component to be tested: create property menu page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify four buttons create house, create apartment, create condominium and back | Admin must be able to view four buttons create house, create apartment, create condominium and back | Admin can see the view four buttons create house, create apartment, create condominium and back | pass |
| **Tst#2** | Click create house button. | Admin must be able to navigate to the create house page which has form for creating house. | Admin can view the create house page. | pass |
| **Tst#3** | Click apartment button. | Admin must be able to navigate to the create apartment page which has form for creating apartment. | Admin can view the create apartment page. | pass |
| **Tst#4** | Click create condominium button. | Admin must be able to navigate to the create condominium page which has form for creating condominium. | Admin can view the create condominium page. | pass |
| **Tst#5** | Click back button | Admin must be able to navigate back to the admin home page | Admin can view the admin home page | pass |

**Test case: 07**

**Project id: Real Estate Application**

**Module name: view/edit property menu.**

**Component to be tested: view/edit property menu page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify four buttons view/edit house, view/edit apartment, view/edit condominium and back | Admin must be able to view four buttons view/edit house, view/edit apartment, view/edit condominium and back | Admin can see the view four buttons view/edit house, view/edit apartment, view/edit condominium and back | pass |
| **Tst#2** | Click view/edit house button. | Admin must be able to navigate to all houses page | Admin can view the all houses page. | pass |
| **Tst#3** | Click view/edit apartment button. | Admin must be able to navigate to the navigate to all apartments page | Admin can view the to all apartments page | pass |
| **Tst#4** | Click view/edit condominium button. | Admin must be able to navigate to the navigate to all condominiums page. | Admin can view the to all condominiums page. | pass |
| **Tst#5** | Click back button | Admin must be able to navigate back to the admin home page | Admin can view the admin home page | pass |

**Test case: 08**

**Project id: Real Estate Application**

**Module name: all houses/all apartments/all condominiums.**

**Component to be tested: all houses/all apartments/all condominiums page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify property table and four buttons preview, edit, delete, and back | Admin must be able to view the whole property table, and with four buttons preview, edit, delete, and back | Admin can view the whole property table, and with four buttons preview, edit, delete, and back | pass |
| **Tst#2** | Click on preview | Admin must be able to navigate to the preview of property page which has whole property details. | Admin can view the property preview page. | pass |
| **Tst#3** | Click on edit | Admin must be able to navigate to the edit property page. Where admin can edit the selected property. | Admin can see the edit property page. And admin can edit the specific property. | pass |
| **Tst#4** | Click on delete | Admin must be able to delete the selected property. | Admin can delete the selected property. | pass |
| **Tst#5** | Click back button | Admin must be able to navigate back to the view/edit property menu page. | Admin can view the view/edit property menu page. | pass |

**Test case: 09**

**Project id: Real Estate Application**

**Module name: user home page.**

**Component to be tested: user home page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify two buttons view property and logout. | User must be able to view the view property and logout button | User can see the view property and logout button | pass |
| **Tst#2** | Click view property button. | User must be able to navigate to the view properties menu. | User can navigate to the view properties menu. | pass |
| **Tst#3** | Click logout button | User must be able to navigate to the home page. | User can navigate to the home page | pass |
| **Tst#4** | Click check property price button. | User must be able to post the property and check the price of the property. | User unable to upload the property. | fail |

**Test case: 10.**

**Project id: Real Estate Application**

**Module name: view properties menu.**

**Component to be tested: view properties menu page.**

**Type of component: screen.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case ID** | **Description of**  **Test case** | **Expected Result** | **Actual Result** | **Pass or Fail** |
| **Tst#1** | Verify five buttons view house, view apartment, view condominium, view property, back. | User must be able to view five buttons view house, view apartment, view condominium, view property, back. | User can see the five buttons view house, view apartment, view condominium, view property, back. | pass |
| **Tst#2** | Click view house. | User must be able to navigate to the user’s All houses page. | user can see the  user’s All houses page. | pass |
| **Tst#3** | Click view apartment | User must be able to navigate to the user’s All apartments page. | user can see the user’s All apartments page. | pass |
| **Tst#4** | Click view condominium | User must be able to navigate to the user’s All condominiums page. | user can see the user’s All condominiums page. | pass |
| **Tst#5** | Click view property. | User must be able to navigate to the user’s All properties page. | user can see the  user’s All properties page | pass |
| **Tst#6** | Click back button | User must be able to navigate to the user home page. | User can see the user home page | pass |