TEST PLAN FOR PROPMART – A REAL ESTATE TRACKING APPLICATION

ChangeLog

Version	Change Date	By	Description
version number	Date of Change	Name of person who made changes	Description of the changes made
1	6-11-2023	Yashasvi Baliyan	Initial

1.Introduction

In the continuously changing real estate industry, digital integration has become critical to maintaining efficiency, transparency, and outstanding client service. Consider a cutting-edge real estate application that will transform the way properties are acquired, sold, and managed. This revolutionary platform not only meets the needs of clients and customers, but it also empowers the agents who facilitate these transactions. The capacity of the company owner to track the activity of its agents in real-time is what distinguishes this application. This real estate software ushers in a new age in the sector, where accountability meets smart technology to ensure a flawless experience for all stakeholders.

Introducing PROPMART, the state-of-the-art real estate software created just for business owners and their representatives. Real estate transactions smoothly integrate with technology in this ever-evolving digital ecosystem, providing a comprehensive solution catered to the particular requirements of property administration, agent collaboration, and client engagement. RealEstatePro distinguishes itself by offering a safe and effective platform for business owners and their representatives to work together without difficulty, guaranteeing a smooth process from client meetings to property acquisition and beyond.

Key characteristics include:

Property Management: Business owners can add, amend, and oversee properties right within PROPMART. Owners can highlight their properties with comprehensive descriptions and excellent photos, which makes it easy for agents to view the most recent listings.

Agent Management: Owners have complete power over the agents in their team. In addition to managing their profiles and allocating particular properties depending on their experience and customer preferences, they may add new agents to the program. This streamlined procedure guarantees the best possible property-client matching while also improving team cooperation.

Property Assignment: The procedure is made simpler with PROPMART. Owners can easily designate properties to particular agents, guaranteeing individualized service for every customer. This tool encourages agents to manage their workloads effectively and with accountability.

Agent Activity Tracking: The ability to trace the activity of every agent affiliated with the firm is at the heart of this application. The owner may monitor everything from client contacts to property viewings, increasing accountability and improving the quality of service given.

Transparency and Accountability: Clients can now have faith in the process, knowing that their transactions are being overseen by the company's owner. This transparency fosters confidence, which is essential in real estate transactions.

Efficient Communication: The program enables agents, clients, and the firm owner to communicate in real time. Messages, updates, and notifications can be shared in real time, keeping everyone informed throughout the transaction.

Performance Analytics: Comprehensive analytics and reports are provided.

Location Tracking: PROPMART's integration of cutting-edge GPS technology enables owners to track and locate their agents in real-time. By ensuring that agents are where they should be, this feature improves client responsiveness and efficiency.

Activity records: In-depth activity records give owners a thorough understanding of the meetings that their agents attended. Owners have access to comprehensive reports on everything from customer contacts to follow-ups, which promotes openness and facilitates well-informed decision-making.

Secure Communication: Data security and confidentiality are top priorities for PROPMART. Sensitive information can be privately communicated with and shielded from unwanted access by owners and agents using encrypted communications within the program.

Extract of the application: With its advanced and user-friendly platform, RealEstatePro redefines real estate administration and is only intended for business owners and their agents. RealEstatePro integrates agent cooperation, efficient property administration, and tracking features to simplify every facet of the real estate industry. RealEstatePro is your reliable partner in attaining unmatched success in the cutthroat real estate market, whether you are an agent looking for optimal client interaction or a business owner seeking flawless property administration. Discover the real estate management of the future, where great customer pleasure is achieved via the combination of efficiency, innovation, and teamwork. Welcome to RealEstatePro, the place where modern real estate endeavors flourish.

1.1 Scope

The scope of the real estate application, where the company owner tracks the activities of its agents, is defined by the specific features and functionalities it offers. Here is the scope of the application:

In-Scope Features:

1.Property Management:

Adding Properties: The application allows the company owner to add new properties to the system. This includes details such as property title, description, images, location, and price.

2. Agent Management:

Adding Agents: The owner can add new agents to the application, providing agent details such as name, contact information, expertise, and territory.

3. Property Assignment:

Assigning Properties:The owner can assign specific properties to individual agents. This assignment could be based on factors like agent expertise, location, or workload.

4. Location Tracking:

Real-Time Location Tracking: The application enables real-time tracking of agents' locations using GPS or other location-based technologies. This allows the owner to monitor agents' movements and activities.

5. Activity Monitoring:

Tracking Agent Activities: The application tracks and logs various activities of agents, such as property viewings, client meetings, and other interactions. This data helps the owner assess agent productivity.

6. Meeting Activity Logs:

Meeting Logs:The application records activity logs related to meetings held by agents. This includes details such as meeting date, time, participants, agenda, and outcomes.

Out-of-Scope Features (Not Included in this Application):

1. End User Expansion: The application is designed exclusively for company owners and agents. It does not extend its services to other categories of users, such as buyers, sellers, or external brokers.

2. Transaction Management: The application does not handle financial transactions related to property purchases or sales. It focuses on agent management and activity tracking.

3. Property Search for Buyers: The application does not provide features for property search or viewing to potential buyers. Its main purpose is to empower company owners in managing their agent workforce effectively.

4. Document Management: Detailed document management, such as storing contracts, agreements, or legal documents related to properties, is not within the application's scope.

5. Third-Party Integrations:The application does not integrate with external services or APIs for purposes other than location tracking, focusing solely on its core functionalities.

6. Analytics and Reporting:While the application tracks activities, it does not include complex analytical features or reporting tools for in-depth business analysis.

1.2 Quality Objective

1.2.1 Quality Objectives:

1. User Experience (UX):-

Objective: Make sure the application offers agents and business owners a simple and intuitive user experience.

Metrics: Average time to perform activities, comments from usability tests, and user satisfaction surveys.

2. Data Security and Privacy: -

Objective: Protect sensitive data while abiding by laws and industry standards, such as property information and user data.

Metrics: Adherence to data protection regulations, accomplishment of security audits, lack of data breaches.

3. Stability and Reliability: -

Objective: Guarantee that the program is available for usage without frequent crashes or outages.

Metrics: Response time during peak usage, number of critical defects reported after release, and application uptime.

4.Performance and Scalability:

Goal: Ensure that the program runs well under varying loads and that it can expand to handle increasing user numbers.

Metrics include response times under varied loads, server resource usage, and the outcomes of scalability tests.

5.Reliability of Activity Monitoring:

Goal: Verify that the movements and interactions of the agents are appropriately reflected in the activity logs and location tracking.

Metrics include the accuracy of activity timestamps and the discrepancy rate between tracked and actual locations.

Testing Objectives:

1.Functional Testing:

Goal: Confirm that all features—assigning properties to agents, tracking locations, creating activity logs, and adding properties—function as planned.

Measures: Defect density and percentage of test cases that pass.

2.Usability Testing:

Goal: Assess the user interface's readability and usability to make sure it's easy to navigate and complete tasks.

Metrics include average time to complete common tasks and user satisfaction scores.

3.Security Testing:

Goal: Find holes in the application's security protocols so that user information is shielded from unwanted access.

Metrics: Total number of security flaws found, amount of time needed to fix serious flaws.

4. Data Integrity and Accuracy:

Goal: Verify that all information supplied into the application—such as agent and property details—is correct and consistently saved and retrieved.

Metrics: accuracy of data retrieval and data validation error rate.

1.3 Roles and Responsibilities

Detail description of the Roles and responsibilities of different team members like
- QA Analyst : Yashasvi Baliyan, Shivendu Mishra
- Test Manager : Ms. Shreela Pareek
- Configuration Manager : Ms. Neha Shukla
- Developers : Shivendu Mishra, Shiva Agrahari, Yashasvi Baliyan
- Installation Team: Shivendu Mishra, Shiva Agrahari

2 Test Methodology

2.1 Overview

1.Agile Methodology:

Iterative Development: Adding functionality incrementally is a common way that real estate systems progress. This method is supported by agile, which permits ongoing input and modifications following each iteration.
User Stories: Agile places a strong emphasis on customer cooperation and user stories. User stories can describe a variety of functionalities for this application, including adding attributes, allocating them to agents, and monitoring actions.
Work is divided into short, two- to four-week work periods called sprints. Every sprint can concentrate on particular features or improvements, enabling the team to produce noticeable outcomes fast.

2.Real User Testing: User Acceptance Testing (UAT) Real users, such as business owners and agents, validate an application's functions against their actual demands through user acceptance testing (UAT). Their input is very helpful in improving the application prior to its official release.

3.Performance Testing:

Stress and Load Testing:To evaluate the performance of the application under stress, simulate a large number of users. Stress testing makes that the system can withstand unforeseen spikes in user activity by pointing out its weak points.

2.2 Test Levels

The testing to be performed is Manual Testing.
The testing to be informed by the developers along with QA and Test Manager.

1.Requirements Analysis:
Specific Requirements: Make sure that every requirement—such as adding properties, allocating properties, managing agents, tracking locations, and generating activity logs—is well-understood and documented.

2.Matrix of Requirement Traceability (RTM): To link every requirement to its appropriate test case, create an RTM. This guarantees that a minimum of one test case covers each criteria.

3.Functional Testing:

Testing, both positive and negative: Test boundary conditions and error scenarios in addition to confirming that each functionality functions as intended (positive testing).

4.Use Case Illustrations: Build and run test cases based on common use cases, such adding a property, designating an agent for it, and monitoring the actions of the agent.

5. Security Testing:

Data Encryption: Ensure that private information, including location data and client information, is encrypted and secured.

Authentication and Authorization: Verify that only authorized users are able to access particular functionality by testing user authentication and authorization systems.

6. Usability Testing:

User Interface (UI) Testing: Verify that all features of the user interface are consistent, easy to use, and intuitive.
Workflow Testing: Verify that the agent management, activity tracking, and property addition workflows are clear and logical.

7. Performance Testing:

Load testing: Verify that the application can manage the anticipated volume of users and transactions by assessing its stability and response time under varied loads.

Scalability Testing: Determine whether the program can grow when more properties, agents, and data are added while maintaining optimal performance.

8. Regression Testing:

Automated Regression Testing: Put essential features through automated testing to make sure that bug patches or new releases don't adversely affect already-existing features.

Rerun Relevant Previous Test Cases: To ensure that the modifications have not resulted in regression problems, rerun pertinent previous test cases following the addition of a new feature or bug patch.

9. Data Integrity Testing:

Database Testing: Check that the information owners and agents enter into the database is appropriately saved and retrieved.
Data validation: To avoid inaccurate or missing entries, make sure that the data (agent information, property details) entered into forms is validated.

10.Logging and Reporting:

Activity Log Verification: Check that the activity logs created following meetings, property assignments, or agent interactions are accurate and comprehensive.
Report Accuracy: Verify the accuracy of reports that are produced using meeting logs and agent activity.

11. Compatibility Testing:

Browser and Device Compatibility: To guarantee responsive design and compatibility, test the application across a range of web browsers and devices.

12.Documentation Verification:

User guides: Confirm that the features and functionalities of the program are appropriately reflected in the user guides and documentation.

13. User acceptance Testing(UAT):

Engage End Users: Perform user acceptance testing (UAT) with business owners and agents to make sure the application satisfies their needs and expectations.

2.3 Test Completeness

For instance, a few criteria to check Test Completeness would be
- 100% test coverage
- All Manual & Automated Test cases executed
- All open bugs are fixed or will be fixed in next release
- All the required data is accurately fetched and OTP is successfully generated and verified.
- 30% Regression tests have been executed, and previously tested features still work as expected after updates or changes.

3 Test Deliverables

Test Cases

Test Cases for Adding Properties:

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
1.	Verify Property Addition	Property should be successfully added to the application	Property appears in the Property List	Pass
2.	Verify Property Addition Validation(Invalid property details or missing information)	Application should display appropriate error messages.	Appropriate error messages are displayed	Pass

Test Cases for Managing Agents:

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
1.	Verify Agent Addition	Agent should be successfully added to the application	Agent appears in the agent list	Pass
2.	Verify Agent Addition Validation(Invalid agent details or missing information.)	Application should display appropriate error messages	Appropriate error messages are displayed	Pass

Test Cases for Assigning Properties to Agents:

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
1.	Verify Property Assignment	Property should be assigned to the selected agent.	Property is assigned to the selected agent	Pass

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
2.	Verify Property Reassignment	Property should be reassigned to the new agent	Property is reassigned to the new agent	Pass

Test Cases for Locating and Tracking Agent Activities:

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
1.	Verify Agent Location Tracking	Application should display the agent's current location on the map	The agent's location is displayed on the map	Pass
2.	Verify Activity Tracking	Application should log and display agent activities.	the activity log displays relevant information about the agent's activities, including date, time, and description.	Pass

Test Cases for Activity Log Retrieval:

S.NO.	Test Data	Expected Output	Actual Output	Pass/Fail
1.	Verify Activity Log Retrieval	Application should display activity logs for the specified date range and agent.	The application displays activity logs matching the specified criteria	Pass
2.	Verify Empty Activity Log Retrieval	Application should display a message indicating no activities found.	The application displays a message indicating no activities were found for the specified criteria	Pass

S.No.	Test Case	Input	Expected Output	Output
1	Verify Property Addition	Property details including title, description, images, and price.	Property should be successfully added to the application	Property appears in the property list
2	Verify Property Addition Validation	Invalid Property details or missing information	Application should display appropriate error messages	Appropriate error messages are displayed
3	Verify Agent addition	Agent's name, contact information, and expertise	Agent should be successfully added to the application	Agent appears in the agent list.
4	Verify agent addition validation	Invalid agent details or missing information	Application should display appropriate error messages	Appropriate error messages are displayed
5	Verify Property Assignment	Selecting an agent and assign a property to them	Property should be assigned to the selected agent.	It is verified that property is assigned to the selected agent
6	Verify Property Reassignment	Reassign a property from one agent to another.	Property should be reassigned to the new agent	It is verified that property is reassigned to new agent.
7	Verify Agent location Tracking	Agent's real time Location	Application should display the agent's current location on the map.	It is verified that agent's location is displayed on the map.
8	Verify Activity Tracking	Agent's activities, such as property viewings or meetings.	Application should log and display agent activities.	It is verified that the activity log displays relevant information about the agent's activities, including date, time and description.
9	Verify Activity Log Retrieval	Select a date range and agent	Application should display activity logs for the specified	It is verified that the application displays activity logs matching

			date range and agent.	the specified criteria.
10	Verify Empty Activity log Retrieval	Select a date range and agent with no activities.	Application should display a message indicating no activities found.	It is verified that the application displays a message indicating no activities were found for the specified criteria.

Decision Table For Admin Login

Test Case	User Name	Password	Expected Output	Actual Output
1	Wrong Admin name	Wrong password	Invalid Admin	Invalid Admin
2	Wrong Admin name	Correct Password	Invalid Admin	Invalid Admin
3	Correct Admin Name	Wrong Password	Invalid Admin	Invalid Admin
4	Correct Admin Name	Correct Password	Login successful	Login Successful

Decision Table For Agent Login

Test Case	User Name	Password	Expected Output	Actual Output
1	Wrong Agent name	Wrong password	Invalid Agent	Invalid Agent
2	Wrong Agent name	Correct Password	Invalid Agent	Invalid Agent
3	Correct Agent Name	Wrong Password	Invalid Agent	Invalid Agent
4	Correct Agent Name	Correct Password	Login successful	Login Successful

4 Resource & Environment Needs

4.1 Testing Tools

Manual Testing: A set of test cases were made using Decision Table and tested by making changes to the backend (Firebase) with the input test data for testing of the App

Postman:
Type: API Testing Tool
Purpose: Postman simplifies the process of developing APIs that allow taking advantage of automated testing. It offers features for designing, testing, and managing APIs, making it easier to ensure that the backend services are functioning as expected.

4.2 Test Environment

Requirement of Flutter-based Android Application:

● Android 5.0+

● Minimum 512 MB RAM, 100 MB Storage

 ● Steady Internet Connection

5   Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test
UAT	User Acceptance Testing
MB	Mega Byte
RAM	Random Access Memory

Testing Faculty: Ms Shreela Pareek

Project Guide: Ms Arti Sharma

Project Submission: Ms Neha Shukla