**Software Requirements Specification**

**For**

**Property Management Mobile App Version 1.1**

**Table of Contents**

**Table of Contents ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Project Scope and Product Features 1

**2. Overall Description 1**

**3. Application Features 1**

**4. External Interface Requirements 2**

4.1 User Documentation 2

4.1 Hardware Interface 2

**5. Wireframe / UI Screens 2**

**6. System Features 12**

# **Introduction**

## **Purpose**

Software Requirements Specification for Property Management Mobile Apps. This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here is a high priority and committed to releasing 1.1.

## **Project Scope and Product Features**

The Software Requirements Specification for Property Management Mobile Apps. It provides intuitive rental tracking for the next generation of mobile landlords and property managers. The users can track their tenants, rent roll, expenses, and more.

# **Overall Description**

Property Management Mobile Apps. It provides intuitive rental tracking for the next generation of mobile landlords and property managers. The users can track their tenants, rent roll, expenses, and more.

# **Application Features**

**3.1** **Tenant Tracking:**   
Manage tenants and their lease terms. Track how much rent is owed by each tenant, when tenants are late on rent, and when lease agreements are ending.  
  
**3.2** **Accounting:**   
Manually add transactions for rent payments, bills, etc..., setup recurring transactions, or import your transactions automatically. Enable double-entry accounting mode and input all accounts affected in a transaction. Reconcile accounts against statements.  
  
**3.3 Task Management:**  
Add to-do lists to track tasks you need to complete on your rentals, such as finding a tenant for a unit that will soon be vacant or fixing a leak in a roof. Assign tasks to vendors and invite them to login to Apps and view/update the tasks assigned to them.  
  
**3.4 Financial Performance:**  
View key performance metrics for each of your properties, such as gross income, operating costs, net operating income, cash gain, and cash on cash invested.  
  
**3.5 Alerts:**   
Alerts will be generated when tenants are late on rent, lease agreements are ending, task deadlines are approaching, and more. These alerts can be delivered to your email or you can just view them on the Apps dashboard.  
  
**3.6 Documents:**   
Upload rental documents to Apps server, such as lease agreements or 3 day notices, and then pull them up on your phone for quick and easy reference.  
  
**3.7 Tenant Interaction:**   
Invite tenants to join Apps, which gives them the ability to view how much rent they owe, submit service/repair requests, contact you, and view their lease agreement.

# **External Interface Requirements**

All components of the software need to be open source. The software must run on the mobile phone like Android operating system. The mobile phone has existing hardware/software constraints. The database used by the software needs to be the same one that is used by the web server.

## **User Documentation**

There are help files within the program for each screen the user sees.

## **Hardware Interfaces**

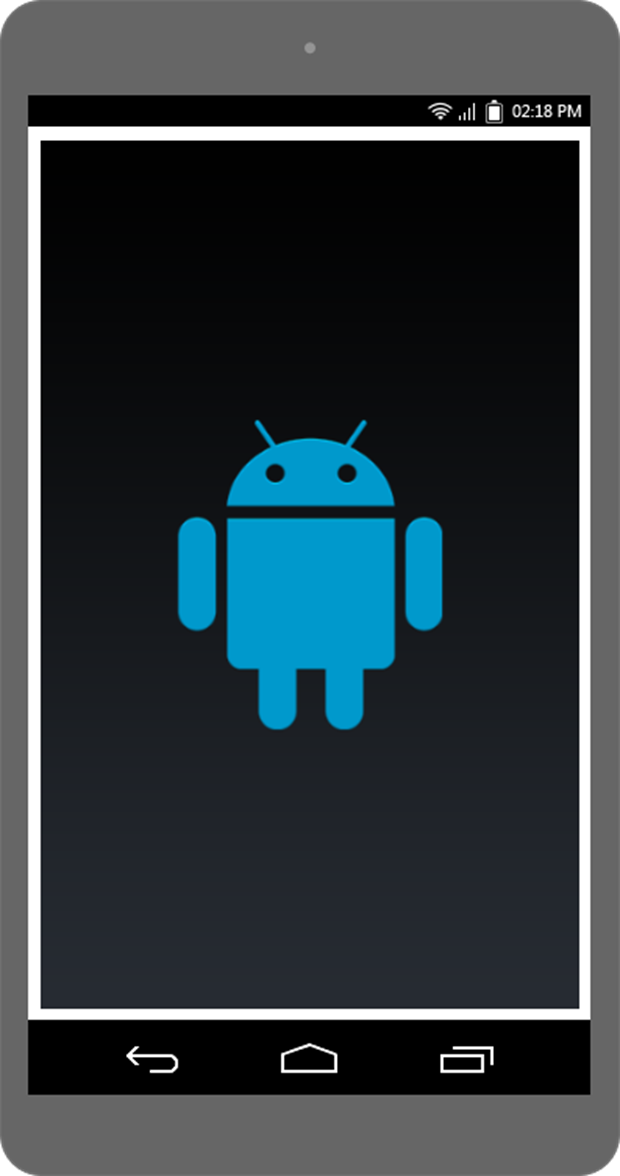
This will be an Android /iOS phone application, and as such will be designed to interface with the hardware present on the Android /iOS phone. In theory the application will be able to run by other devices that can emulate the Android/ iOS, but this will not be a consideration during design.

As this is a mobile device, it will be using the Android network to connect to the internet, which will allow it to communicate with the database servers.

# **Wireframe / UI Screen**

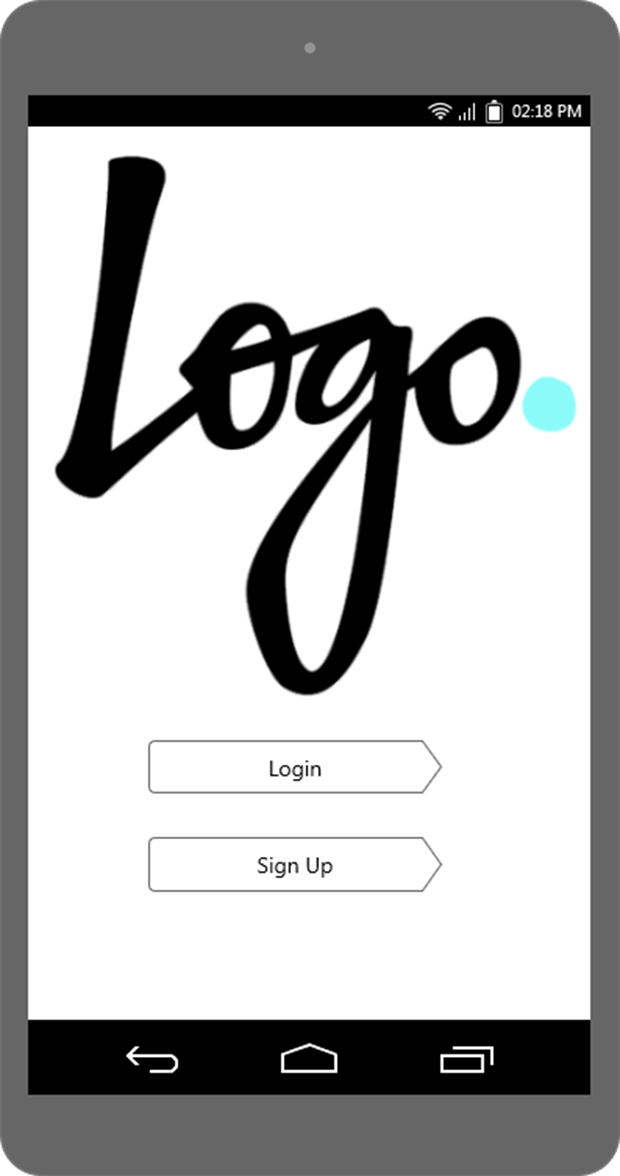
## **Splash Screen**

Android splash screen are normally used to show user some kind of progress before the app loads completely. Some people uses splash screen just to showcase their app / company logo for a couple of second.



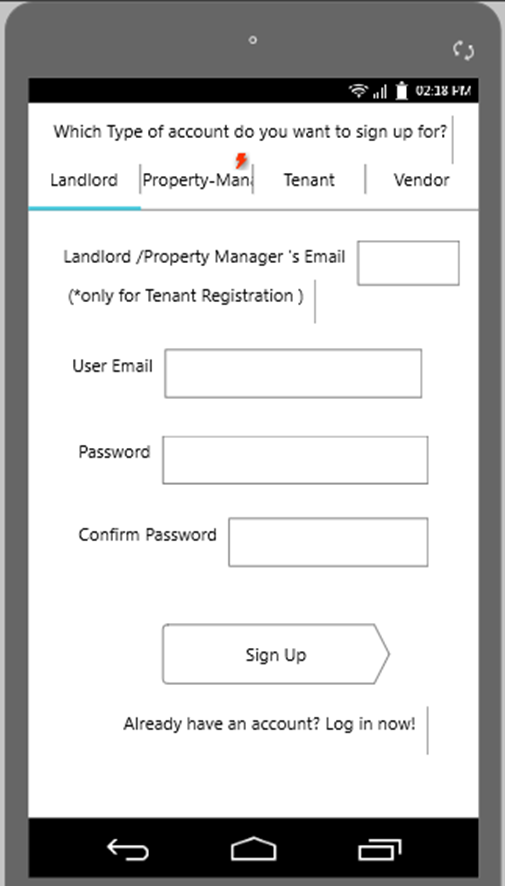
## **Login Registration Choice Screen**

In this screen user can go **for Login and Registration screen**



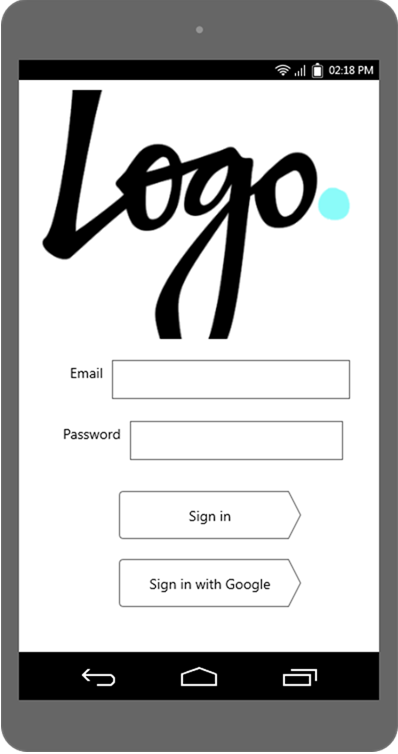
## **Registration Screen**

User Registration screen for registering user with app. For different user use different registration screen select from Tabs i.e. Landlord or Tenant.



## **Login Screen**

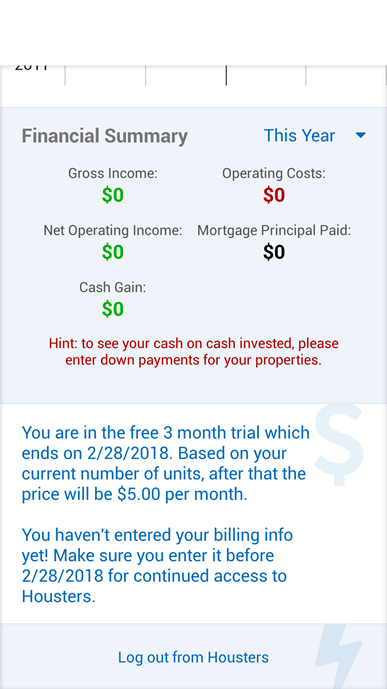
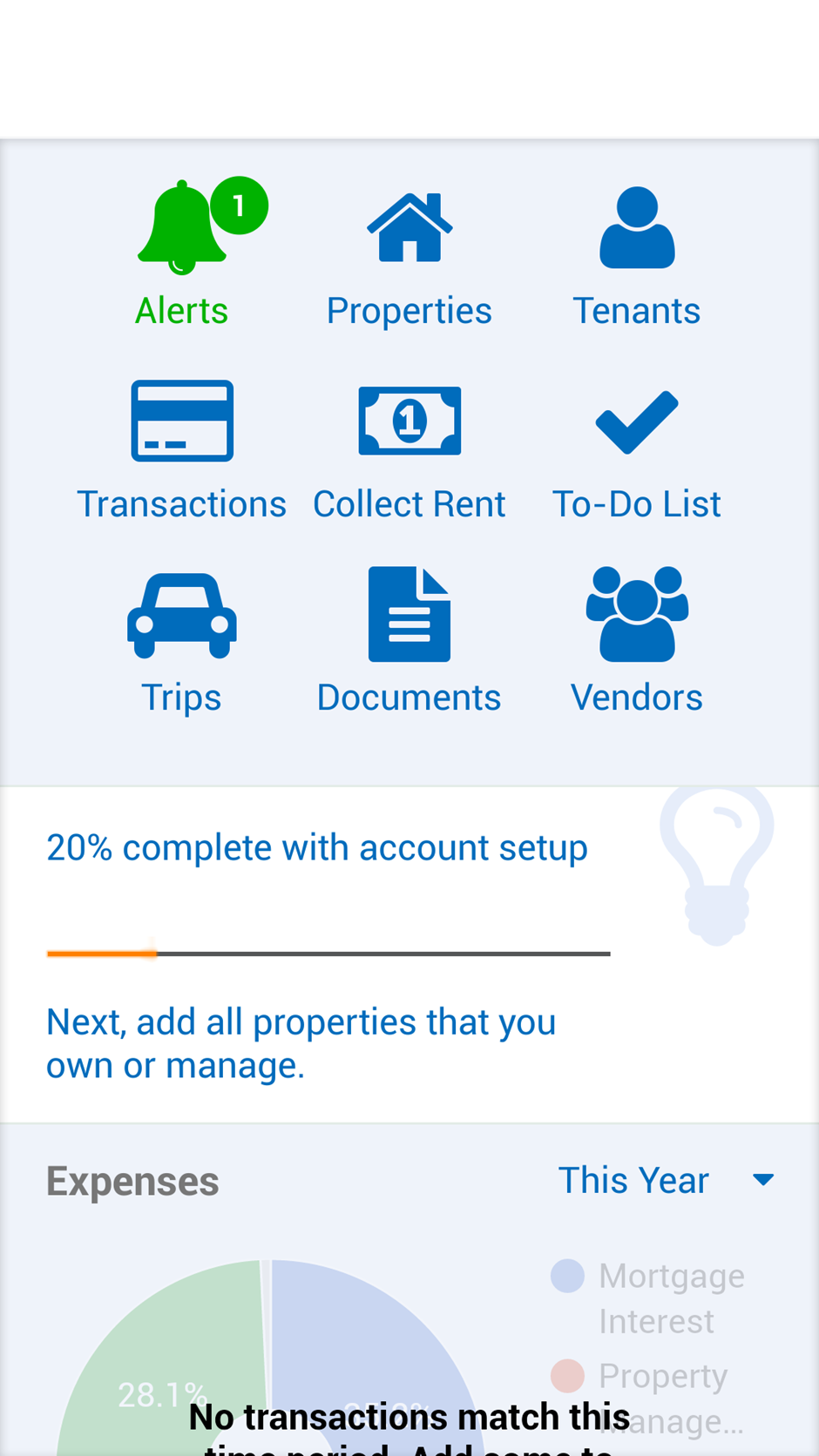
Authentication /login screen for candidate who can manage property/rent.



## **Home Screen**

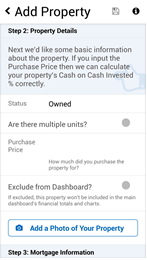
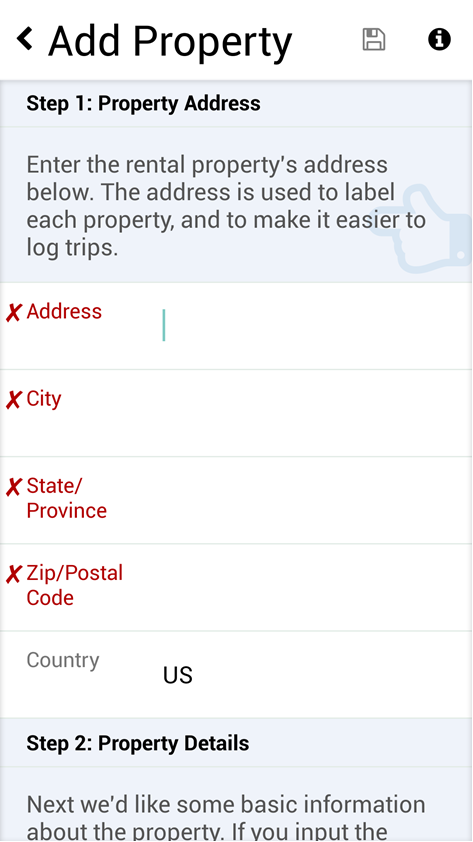
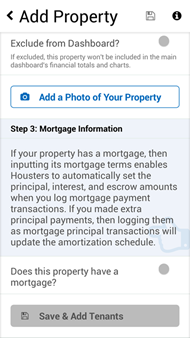
This is Home screen design use scroll view for the complete view we have split the screen into three screens.

In this screen, a user can go inner screens as well as see expenses with chart view and also see financial summary by year or months.

In 

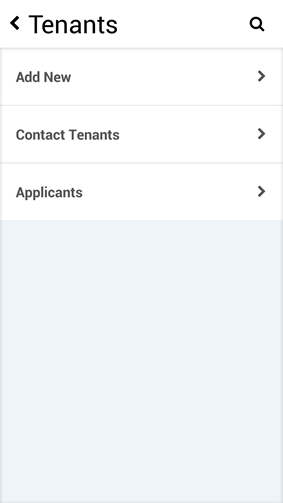
## **Property Screen**

In this screen user can see their property list or can add property.



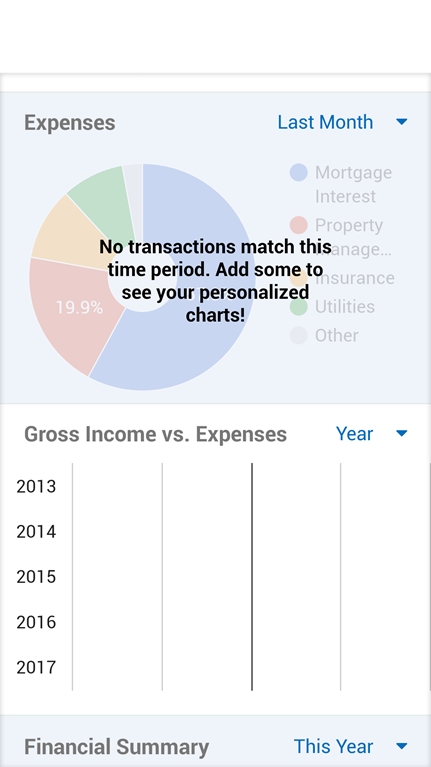
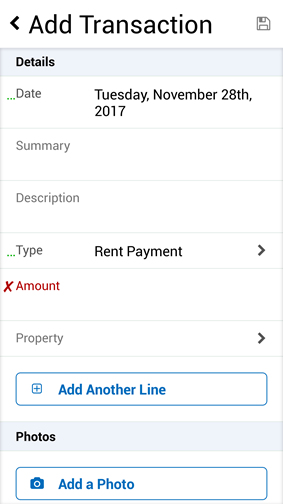
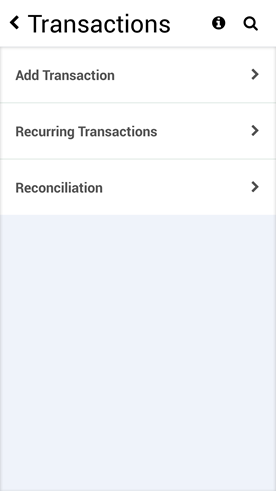
## **Tenants Screen**

In this screen user can manage tenant’s information.



## **Transactions Screen**

In this screen user can manage Transaction’s information.



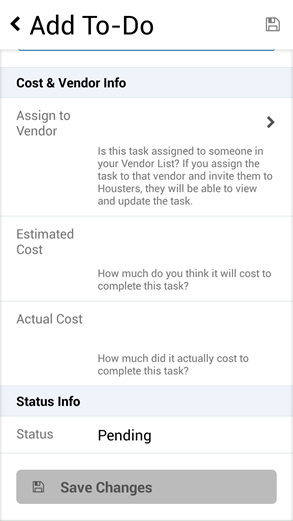
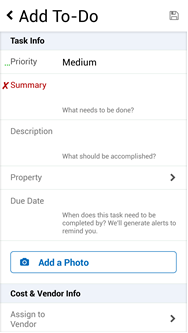
## **Collect Rent Screen**

In this screen user can manage rent’s information.

# 

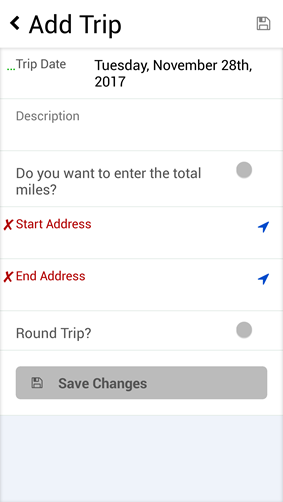
## **To Do List Screen**

In this screen user can manage to do task for property.



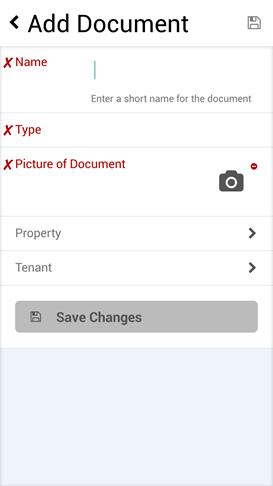
## **Trip Screen**

In this screen can manage Trip’s information.



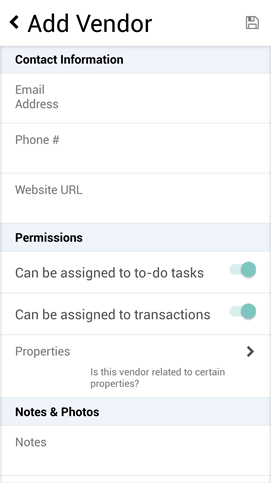
## **Document Screen**

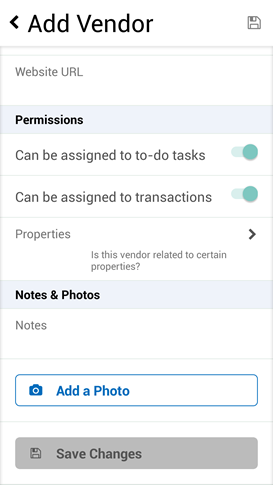
In this screen can manage property or tenant’s documents.



## **Vendor Screen**

In this screen can manage Vendor’s information & given task.





# **System Features**

In this subsection, we will examine the features and entire flow of the system in detail by categorizing them according to their functionality. For each of the feature, we will give an introduction, purpose, flow diagram and the sample output window. Introduction part will give basic background information about the feature. After that, we will show a diagram for the feature representing flow of events.