PES UNIVERSITY, BANGALORE

Software Requirements Specification

HOMELY

An Online Rental Application

SL.NO	NAME	SRN
1	Nilkant	PES2UG20CS530
2	Pradhyumna	PES2UG20CS533
3	Shreepathi	PES2UG20CS553
4	Manik	PES2UG20CS576

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Pradhyumna Nilkant Manik Shreepathi	The final version of the SRS document has been drafted with all the requirements being incorporated into the document.	

.

Introduction

1.1 Purpose

The purpose of our product is we provide two user views.

- One can register their PG, rental rooms or any stays available.
- Another person who is new to the city can see and explore the options available in that locality.

The product's main aim is to provide all options to stay in that locality which new people to that locality can't explore.

1.2 Intended Audience

- Tenant - User who is new to that locality and wants to explore the stays available
- Owner – User with the place they want to rent out.

1.3 Product Scope

Our main focus is on students and workers who are shifting to a new city or new locality and who want a place to stay like PG's, single independent rooms or BHK many more options they can't physically visit and explore the all places of stay because of many constraints like time many unknown stays available at that place.

But with our product, they can explore the many stays available in that locality and can compare among options available based on their interest.

In many cases, they are fooled by brokers who don't know about the prices of the stays around that locality. but with this product, they can compare all options and verify themselves if is it worth that.

And in another view, many owners who don't have a good reach can register their stays and services such as in the case of PG's – food, wife, laundry, and in independent stays about water and many other things and let users know about amenities.

1.4 References

Following are the references for our project

- **NO broker**
- **Google maps**

1.5 Document Conventions

Formatting Conventions:

- The font style for the headings of each section is Arial Bold and the font size is 18.
- The font style for the headings under each section is Arial Bold and the font size used is 12.
- For the remainder of the document, the font style is Arial and the font size is maintained at 11.
- Italics have been used to indicate comments.
- The text is single-spaced and margins are maintained at 1'' separation.

2. Overall Description

2.1 Product Perspective

ADMIN MODULE:

This Admin module contains the complete details about registered users and rooms. Some modules include login, user view, and add/view rooms.
customer Login is used to create a login and sign in to the login. The user view contains the user details.

The sub-modules are:

1. LOGIN
2. USER VIEW
3. ADD/VIEW ROOMS

USER MODULE:

User Modules contain every user detail who registers as the user. The people who want to search for stays to rent. They haven't registered as a user and log the user can check the room details and availability of the room which is in that locality. Instead of walking around and asking for a vacant house. The details about the rooms can view in this module.

1. REGISTER
2. LOGIN
3. VIEW ROOM STATUS IN THAT AREA.

RENTAL OWNER MODULE:

The rental owner module contains the availability of the room and the user can view the room.

2.2 Product Functionality

These are the major functionalities of the product:

- Provide different views to the user depending on his/her role.
- Add the stays and services available.
- View all stays in that area available.

2.3 Users and Characteristics

- Admins
- Tenants
- Rental Owners

The various users that we expect the software to be used by are:

1.	ADMINS	Developing, Verification, Maintenance of the application
2.	USER	A user looking for a stay.
3	Rental Owners	The user who wants to register about their stay and services.

2.4 Operating Environment

An android device with an android version greater than 10.

2.5 Design and Implementation Constraints

We have to design different pages for different types of users such as admin, users' owners' and tenants.

2.6 Assumptions and Dependencies

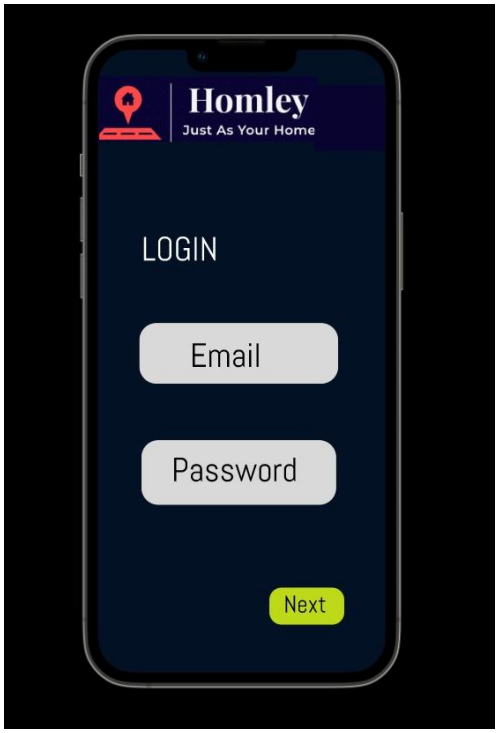
Users should have an android device that supports the application.

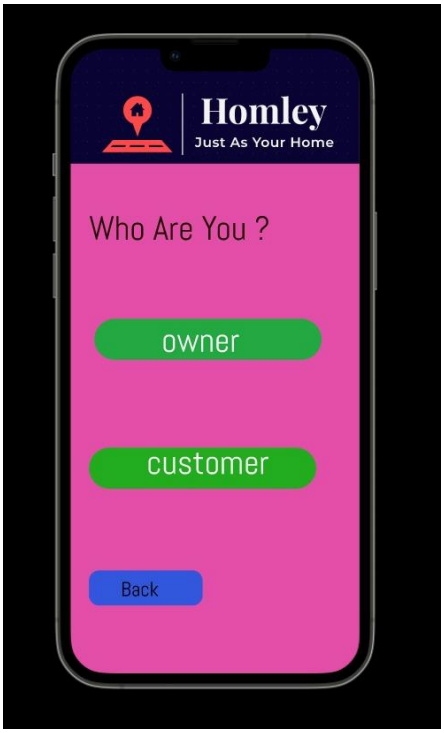
3. External Interface Requirements

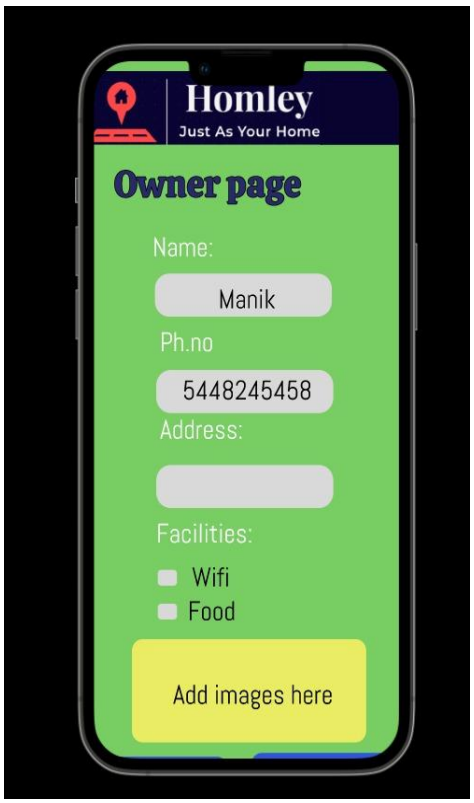
3.1 User Interfaces

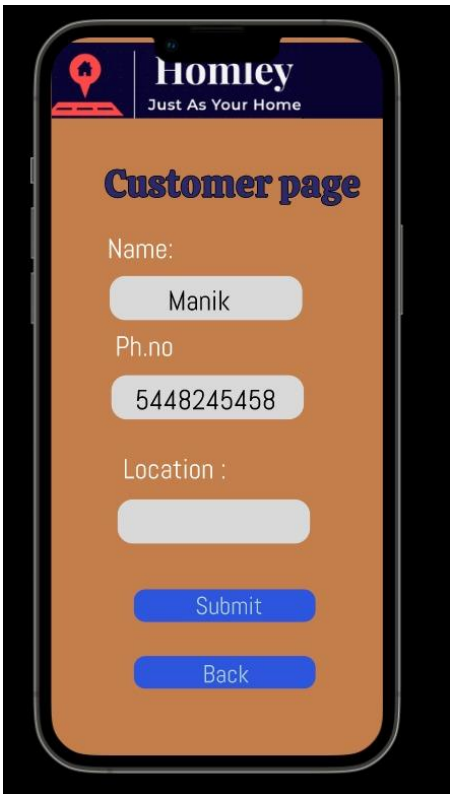
- The user interface is simple, straightforward, and easy to use. One may easily find the stays available in that area.
- If a user is new to the software, then use the r can register for a new account to gain access to the app services.
- Owners /Landlords can add their rooms/pg. easily.

Sample Screenshots:









3.2. Hardware Interfaces

No Specific hardware requirements. Compatible with any android device.

3.3. Software Interfaces

Android OS (Android version greater than 10).

3.4. Communications Interfaces

The software cannot be accessed without an android device.

4. Functional Requirements

- Owners are assigned a unique ID on Registration for verification.
- Owners can list their Rooms/PGs.
- Owners can add Features like the number of rooms available, Amenities like Meals, Internet Services, housekeeping, etc.
- Users can specify their location and access Available nearby rooms instantly.
- Users get complete information regarding the place, facilities specified by the owner, etc.
- For a better experience, users can check reviews of the place.
- Additionally, users can provide feedback on the application.
- Users can also make online payments via Card, or UPI.

3.3. Behavior Requirements

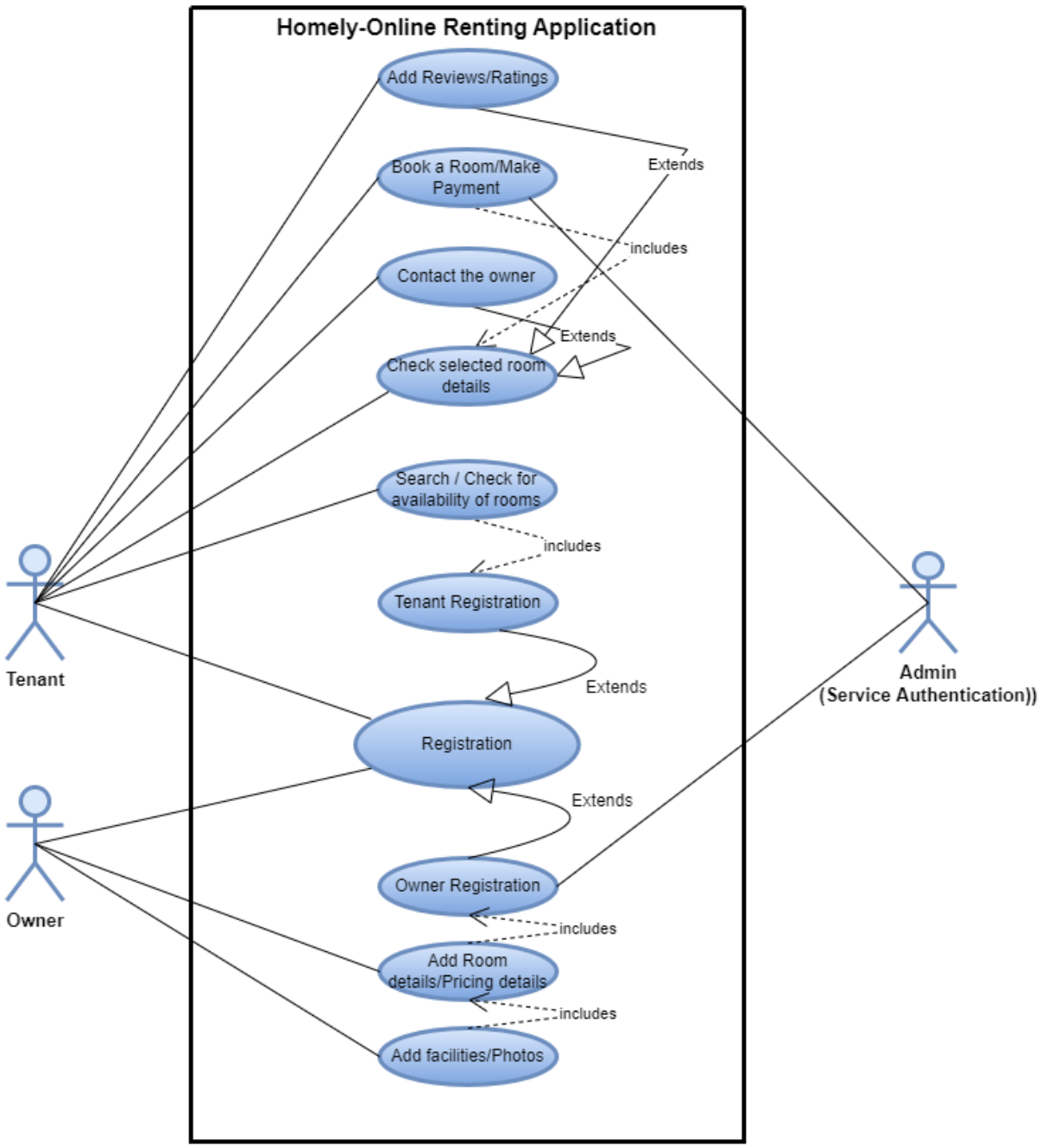
3.3.1 Use Case View

3 actors have been identified for the project and each of them is shown with their own set of use cases.

The actors are depicted as stick figures in the use case diagram.

An overall system view of the software is depicted with all the actors and their corresponding use cases put together.

All the events would have to proceed with a Login of the user (Owner/Tenant). The other events are self-explanatory as shown in the use case diagram.


Homely-Online Renting Application
Add Reviews/Ratings
Book a Room/Make Payment
Extends
includes
Contact the owner
Extends
Check selected room details
Search / Check for availability of rooms
includes
Tenant Registration
Extends
Registration
Extends
Owner Registration
includes
Add Room details/Pricing details
includes
Add facilities/Photos
Tenant
Owner
Admin
(Service Authentication))

4. Other Non-functional Requirements

4.1. Performance Requirements

- Real-time data access (Rooms. PGs listed by owners in the database are updated in real-time).
- Instant online booking by the tenant.
- Multiple users are supported.

4.2. Safety and Security Requirements

- User (Owner/Tenant) has to log in through a secure ID.
- Transaction (Booking info) is maintained for better crash recovery and security
- Privacy of owner and tenant is maintained

4.3. Software Quality Attributes

The application will follow **MVVM architecture** there by following the separation of concerns. The database layer is completely isolated from the UI making Maintenance and testing easy. **The software can be used on android devices with versions greater than 10**. The system can further be modified with additional features.

5. Other Requirements

5.1 Requirements Elicitation:

Requirements were analyzed and gathered by interacting with students living in rooms/pgs. Owners' perspective was considered by interacting with owners/landlords

The following are the requirements finalized after interaction with users and the team

- Need an application to fetch the gap between the owners and tenant
- Simple and easy-to-use application
- Major purpose of the application is to provide details about nearby rooms to the user
- Application should provide all the details of the amenities provided by the rooms/pg.
- Application should enable online booking of the rooms/pg.'s.

Economic feasibility:

Attempts to weigh the cost of developing and implementing the approximate cost of building a new system.

Schedule feasibility:

The proposed project can be completed during the assigned time. Hence it is schedule feasible.

5.2 Technical Feasibility:

Technical feasibility:

For the implementation of the software, the technical resources needed were estimated.
The current solution to the software was decided based on

- The complexity of the technical resources needed.
- Team member's prior experience with the technology.
- Ease of learning the implementation tool.

7.2. Technology Requirements

➢ Front End: Android studio with Kotlin

Android Studio is the official[8] integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.[9] It is available for download on Windows, macOS, and Linux-based operating systems.[10] It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

➢ Back End: MySQL, phpMyAdmin

phpMyAdmin is a free and open-source administration tool for **MySQL** and **MariaDB**. As a portable web application written primarily in **PHP**, it has become one of the most popular **MySQL** administration tools, especially for web hosting services.