DECLARATION BY THE CANDIDATES

We hereby declare that the project report entitled "**Far From Home**" submitted by us to Chandubhai S. Patel Institute of Technology, Changa in partial fulfilment of the requirement for the award of the degree of **B.Tech** in Computer Engineering, from U & P U. Patel Department of Computer Engineering, CSPIT/FTE, is a record of bonafide CE345 Software Group Project-2 (project work) carried out by us under the guidance of **Prof. Martin Parmar**. We further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Kevin Patel (17CE074)
Namrata Patel (17CE078)
Nisarg Patel (17CE080)
Siddharth Patel (17CE086)
Meet Ramaiya (17CE090)
Hussain Sadikot (17CE095)

Prof. Martin Parmar
Assistant Professor
U & P U. Patel Department of Computer Engineering,
CSPIT/FTE, CHARUSAT-Changa.

ABSTRACT

"Far From Home", an Android Mobile Application is a Rental System developed mainly for students, employees and other professionals who are living as PGs or on rent and also for the property owners who are searching for a platform to post their property. It will work mainly to book flats or houses on duration of months or year as per the requirements of the User. The sole purpose of this application is to achieve complete transparency between the owner and the tenant in the entire process of renting by removing the role of a middleman – a broker. This will help the owner to get verified tenants and the tenants get a wide range of property options as per their requirements.

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

PROJECT SUMMARY

This report discusses about Far From Home – a House Rental System , an Android Mobile Application which will help students, employees and other professionals who are living as PGs or on rent and also for the property owners who are searching for a platform to post their property. It also brings complete transparency between the customer and the owner removing the middlemen.

PURPOSE

Many students, employees and other professionals find difficulties in finding house on rent. Also there are property owners who are searching for a platform to post their property. This is the best platform for of them. Also it is such system which excludes the work of middlemen, i.e. the broker and thus, brings transparency in the process.

OBJECTIVE

The Objectives of the Application are:

- Provide Owners a platform so that they can add their property along with images and other information.
- Let users search for house deals using sorting and filters and also add it to favorites.
- The application provides list of the house deals.
- The system provides options to delete/edit/view house deals and to contact owner.

SCOPE

The Things it can do

- The system provides login/signup facilities to the users including OTP verification.
- The system provides the members with the option of forgot password and also delete user profile.
- Owners can add the images of their property along with other information.
- The user can search for house deals using sorting and filters and also add it to favorites.
- The application provides list of the house deals.
- The system provides options to delete/edit/view house deals and to contact owner.

CHAPTER 2

PROJECT MANAGEMENT

PROJECT PLANNING

Project Development Approach and Justification Approach

In our project(Far From Home) we use Agile software process model for mobile application development. In earlier days Iterative Waterfall model was very popular to complete a project. But nowadays developers face various problems while using it to develop a software. The main difficulties included handling change requests from customers during project development and the high cost and time required to incorporate these changes. To overcome these drawbacks of Waterfall model, in the mid-1990s the Agile Software Development model was proposed. The Agile model was primarily designed to help a project to adapt to change requests quickly. So, the main aim of the Agile model is to facilitate quick project completion. To accomplish this task agility is required. Agility is achieved by fitting the process to the project, removing activities that may not be essential for a specific project. Also, anything that is wastage of time and effort is avoided.

Project Effort and Time, Cost Estimation

Using Functional Point Matrix

- No. of External Inputs: 22
- No. of External Outputs: 25
- No. of External Inquiries: 24
- No of External File Interface: 2
- No. of Internal Logic Files: 24

UFP = (No. of External Inputs)*4 +(No. of External Outputs)*5 +(No. of External Inquires)*4 + (No. of Internal Logic Files)*10 +(No. of External File Interface)*10
= 22 * 4 + 25 * 5 + 24 * 4 + 24 * 10 + 2 * 10
= 569

Degree of Influence (DI) = 50
Technical Complexity Factor (TCF) = (0.65 + 0.01 * DI)
= 1.15
Functional Points (FP) = UFP * TCF
= 569 * 1.15
= 654.35
= 655
Line of Code (LOC) = 20 * FP
= 20 * 655
= 13,087

Using COCOMO Model for Semidetached
Effort=3.0 $(KLOC)^{1.12}$ PM
= 3 * $(13)^{1.12}$ PM

= 53 PM

Development Time = 2.5 $(\text{Effort})^{0.35}$ M

= 2.5 * $(53)^{0.35}$ M

= 10 M

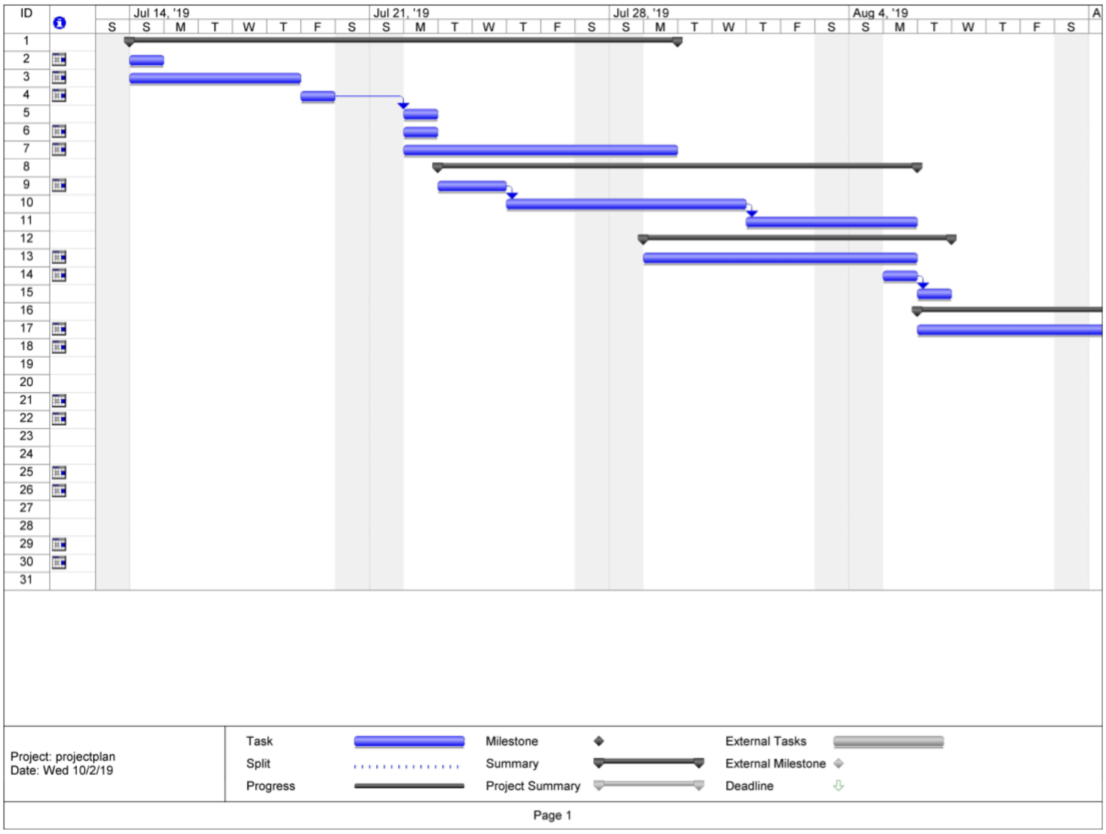
Roles and Responsibilities

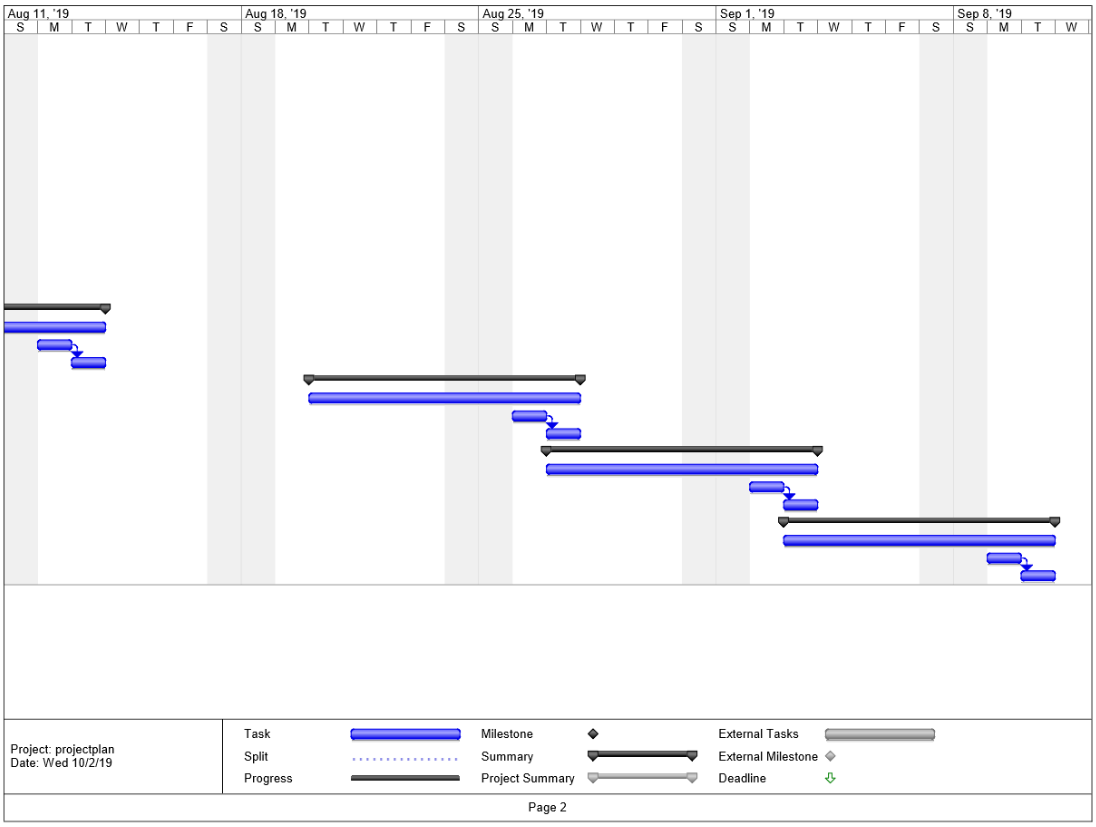
Table 2.1 Roles and Responsibilities

MEMBER NAME	***Responsibility***
Kevin Patel	Project Manager, Design Architect
Siddharth Patel	UI/UX Designer , Developer
Nisarg Patel	Developer
Meet Ramaiya	Developer
Hussain Sadikot	Developer, Documentation
Namrata Patel	Tester, Documentation

PROJECT SCHEDULING

Fig 2.1 Gantt Chart


ID
Jul 14, '19
Jul 21, '19
Jul 28, '19
Aug 4, '19
A
S S M T W T F S S M T W T F S S M T W T F S S M T W T F S
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
Project: projectplan
Date: Wed 10/2/19
Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline
Page 1


Aug 11, '19
Aug 18, '19
Aug 25, '19
Sep 1, '19
Sep 8, '19
S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W
Project: projectplan
Date: Wed 10/2/19
Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline
Page 2

CHAPTER 3

SYSTEM REQUIREMENTS STUDY

USER CHARACTERISTICS

The users of the application are the owners who wish to post their properties on rent and the tenants who wish to stay on rent. Both the parties should have the basic knowledge of the use of mobile and internet which may help them to use the application with ease.

HARDWARE AND SOFTWARE REQUIREMENTS

Developer-side Specifications

Hardware Requirements:

- Microsoft Windows 7/8/10 (32-bit or 64-bit)
- 3 GB RAM minimum, 8 GB RAM recommended (plus 1 GB for the Android
- Emulator)
- 2 GB of available disk space minimum, 4 GB recommended (500 MB for IDE plus
- 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

User-side Specifications

Hardware Requirements:

- Dual Core CPU
- Minimum: Memory 512MB RAM (dependent on modelsize)
- Multitouch supported display.
- Display between 4.7 and 6 inches.
- Internet Drivers

Software Requirements:

- Android OS

ASSUMPTIONS AND DEPENDENCIES

- Assuming that user do have the basic knowledge of operating the internet and access it. The administrator is expected to be familiar with the interface of the tech support system.
- The software is highly dependent on the type of the operating system i.e. it is an android application so it will not work with phones having different operating systems.

CHAPTER 4
SYSTEM ANALYSIS

STUDY OF CURRENT SYSTEM

There are 5 competitions that we have on Android platform where we are going to launch our application but our app is unique from them, as these apps are designed for huge places that take place over all of the city, state or country, while we are targeting a specific audience like places nearby college campus as in to take advantage of the small community. All these apps have high-level backend and UIs but our target audience is totally different from them just the application is the same. Therefore, there is no competition to us in publishing this application as it is limited to the CHARUSAT campus.

Similar apps are:

- 99Acres
- MagicBricks
- NoBroker
- Housing – Property Search & Real Estate
- FlatChat

PROBLEM AND WEAKNESS OF THE CURRENT SYSTEM

- Searching is not supported.
- This is only applicable for android users.

REQUIREMENTS OF NEW SYSTEM

Functional Requirements

F1: Authentication and Authorization

F1.1 User login/signup along with the OTP verification.

- Input: Username and Password for login or Username/E-mail, Password, Mobile Number for signup
- Output: Valid user or not valid user for login or new user created for signup along with OTP verification.

F1.2 Forgot Password and Reset Password in case the user forgets the password.

- Input: Selects Forgot Password and Reset Password option and inputs new password.
- Output: New password is set.

F2: Edit or Delete User Profile:

F2.1 Edit User Profile

- Input: User makes changes in his/her profile data.
- Output: Profile is edited.

F2.2 Delete User Profile

- Input: Select delete account.
- Output: User Profile is deleted.

F3: Provide comprehensive details:

F3.1 System lists out all the details of house.

- Output: List of houses is displayed.

F3.2 Add houses to favorites

- Input: User login and select add to favorites
- Output: House added to favorites

F3.3 Search houses using sort(basis on rent – low to high/high to low, post time – newest first, recently updated) and filter (city, apartment type, BHK type, property size, no. of floors)

- Input: Select any sort or filters.
- Output: List of houses appears according to the sort or filters applied.

F3.4 Owner can edit/delete the house deal

- Input: Edit or delete the property.
- Output: Does not display that property. Property is removed from the database.

F4: Add property:

F4.1 User can add the property along with its images.

- Input: Add the property and its images.
- Output: Property is added and its images are displayed.

F4.2 Show details of the property

- Input: Select any property.

- Output: All types of information related to the location, facilities, amenities, cost, space of property is provided by the owner. The owner can also add the location using the Google Maps.

F5: Contact between the owner and tenant:

- The owner and the tenant can have direct contact between each other.
- There is no middleman required in the process.

Non Functional Requirements

Performance Requirements

The system must be efficient to handle traffic. For now, in our system 10 user can simultaneously use the application without any error.

Safety Requirements

There are two major types of software f failure which may cause possible loss or damage to system: Software logic errors and Software support errors.

Software logic errors are often a result of the programmer making errors in the coding, whether it is simply a mistake on their part, or an incorrect set of requirements they are following. In addition, it may have been a mistake made in the design phase which follows through into the implementation of the system.

Software support errors are linked to the software being used to create the program. Perhaps these errors are from the compiler, an external library being used, or even the programming language.

We have made some safety requirements trying to solve the problems above. They are:

1) Analyze before coding
2) Be careful when coding
3) Do large amount of tests after coding

Security Requirements

At the time of registration, as they will register a one-time password will be send to the registered number and the user will be allowed to proceed further only after he/she verifies that code. The password generated by the user will be encrypted. There is full assurance of no data loss.

Software Quality Attributes:

Usability:

Usability defines how difficult it will be for a user to learn and operate the system.

The system should satisfy a maximum number of customer needs.

No predefined skills are required to use this system. User can easily interact with the system.

Most of the tasks a user can complete without any help. It does not have complex design so any user can easily interact.

Flexibility:
Flutter is an open-source mobile application development framework. It is cross-platform so can be used for both Android as well as iOS.

Availability:
There are no restrictions for the use of the system. The system is open for all interested seekers. The user must have access to any hardware device with Android support and internet connection.

Reliability:
Reliability defines how likely it is for the software to work without failure for a given period of time. The database update process must rollback all related updates when any update fails.

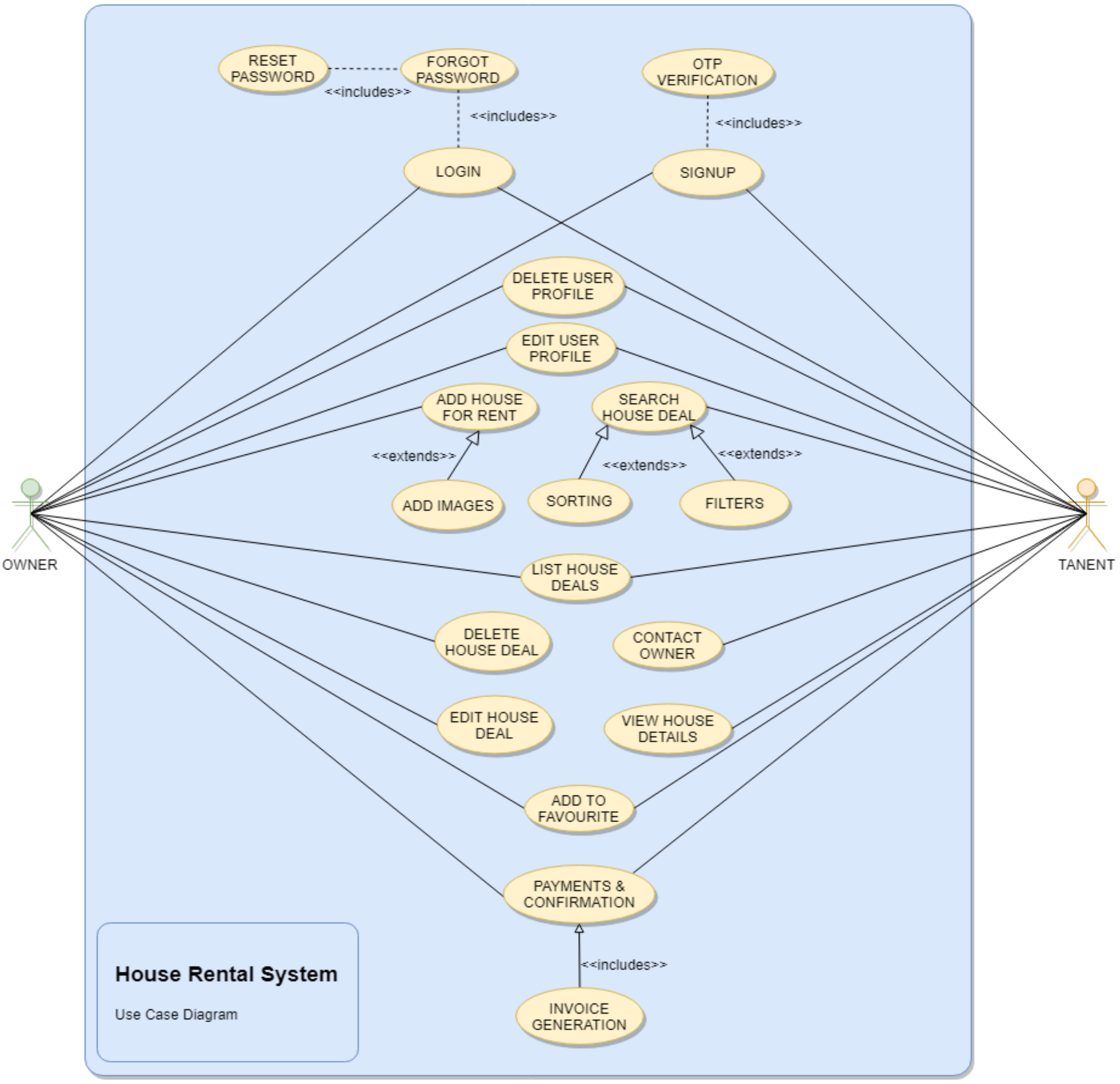
Security:
The system shall use secure sockets in all interactions that include any confidential user information. The system shall log out all the users after a period of inactivity.

Performance:
Performance is a quality attribute that describes the responsiveness if the system to various user interactions with it. The front-page load time must be no more than 5 seconds.

DESIGN CONTROL

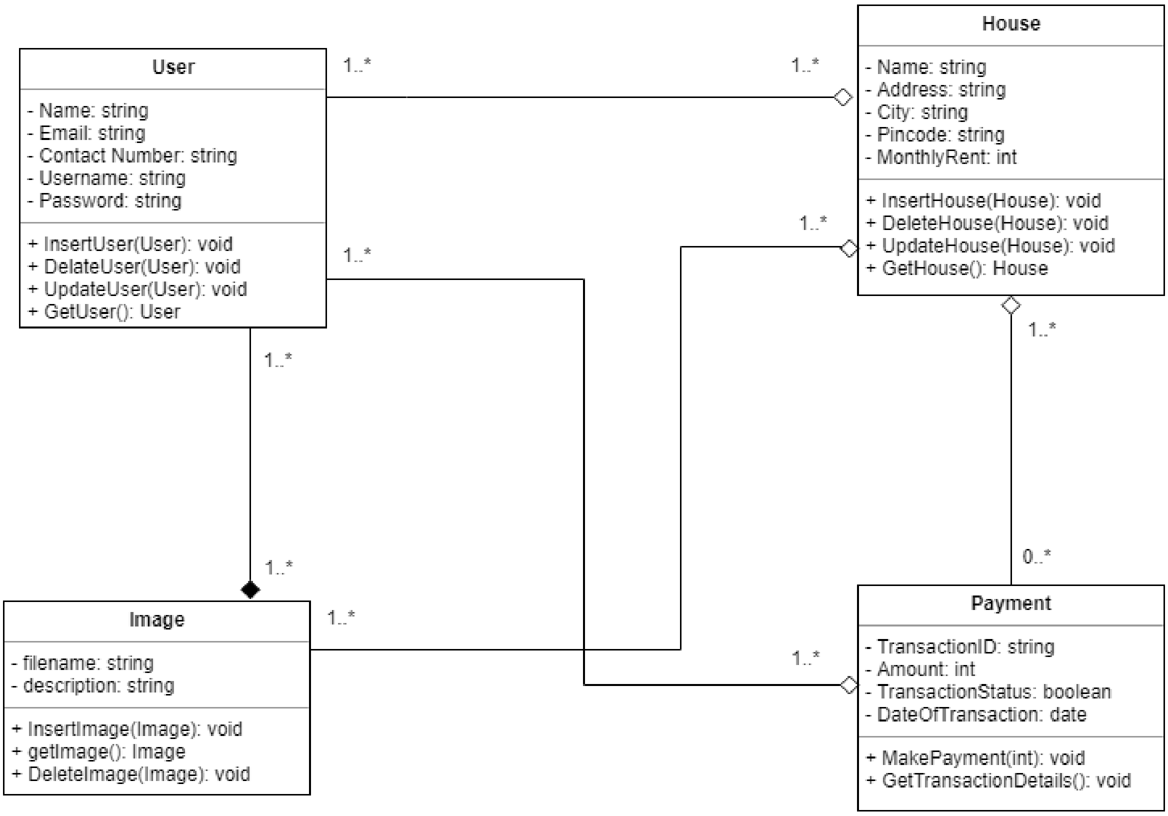
Use Case Diagram

Fig. 4.1 Use Case Diagram



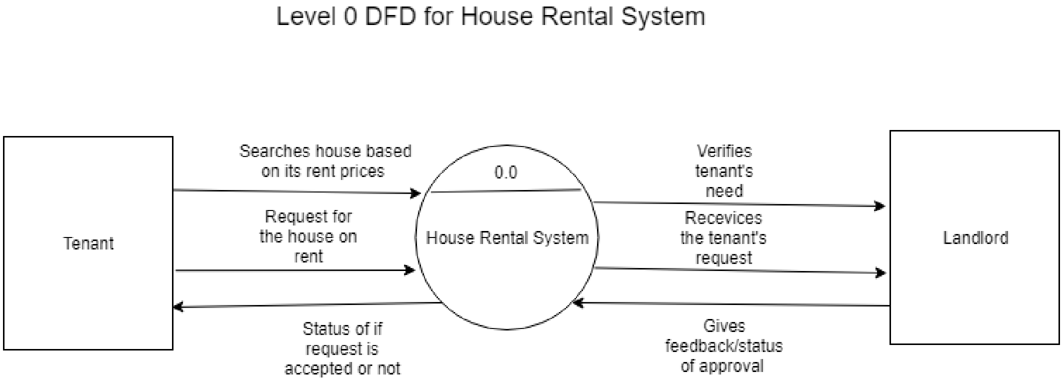
Class Diagram

Fig. 4.2 Class Diagram

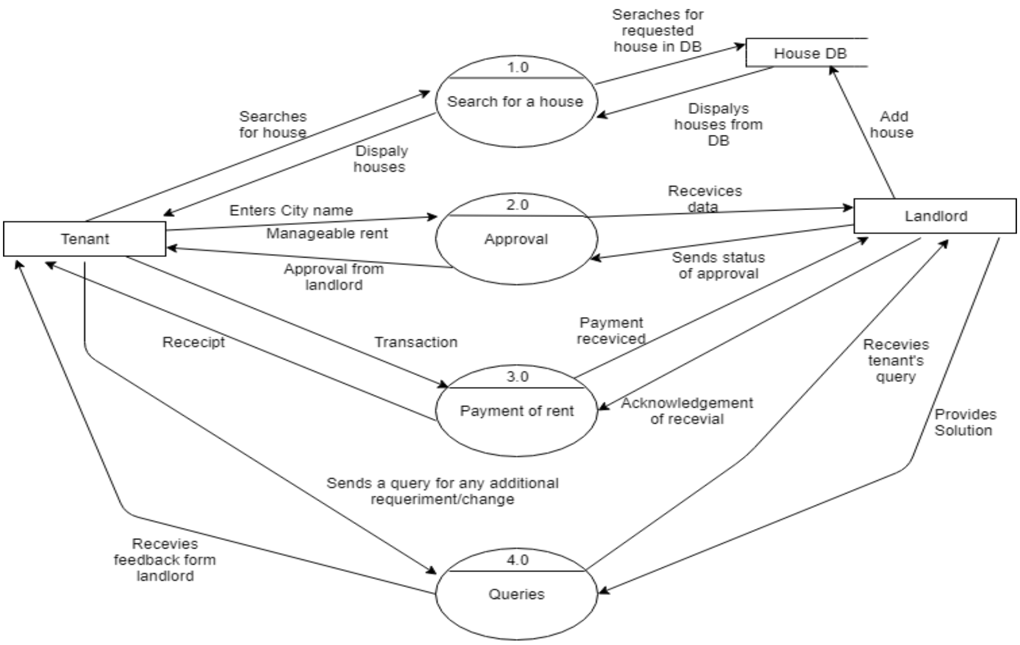


Data Flow Diagram (1 And 2 Level Or Higher)

Fig. 4.3 Data Flow Diagram (0 and 1 level or higher)


Level 0 DFD for House Rental System
Tenant
Searches house based on its rent prices
Request for the house on rent
Status of if request is accepted or not
0.0
House Rental System
Verifies tenant's need
Recevices the tenant's request
Gives feedback/status of approval
Landlord

Level 1 DFD For House Rental System



CHAPTER 5

TESTING

TEST PLAN

A TEST PLAN is a document describing software testing scope and activities. It is the basis for formally testing any software/product in a project.

Test Plan Guidelines:

- Make the plan concise. Avoid redundancy and superfluousness. If you think you do not need a section that has been mentioned in the template above, go ahead and delete that section in your test plan.
- Be specific. For example, when you specify an operating system as a property of a test environment, mention the OS Edition/Version as well, not just the OS Name.
- Make use of lists and tables wherever possible. Avoid lengthy paragraphs.
- Has the test plan reviewed a number of times prior to baselining it or sending it for approval? The quality of your test plan speaks volumes about the quality of the testing you or your team is going to perform.
- Update the plan as and when necessary. An out-dated and unused document stinks and is worse than not having the document in the first place.

TESTING STRATEGY

A test strategy is an outline that describes the testing approach of the software development cycle. It is created to inform project managers, testers, and developers about some key issues of the testing process.

- Black Box Testing:

  A test strategy is an outline that describes the testing approach of the software development cycle. It is created to inform project managers, testers, and developers about some key issues of the testing process.

  In BlackBox Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program.

  How to do BlackBox Testing?

  Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially, the requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them.
- Tester determines expected outputs for all those inputs.

- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.
- Defects if any are fixed and re-tested.
- White Box Testing:

  White Box Testing is defined as the testing of a software solution's internal structure, design, and coding. In this type of testing, the code is visible to the tester. It focuses primarily on verifying the flow of inputs and outputs through the application, improving design and usability, strengthening security. White box testing is also known as Clear Box testing, Open Box testing, Structural testing, Transparent Box testing, Code-Based testing, and Glass Box testing. It is usually performed by developers.

  What do you verify in White Box Testing?

  White box testing involves the testing of the software code for the following:

  - Internal security holes
  - Broken or poorly structured paths in the coding processes
  - The flow of specific inputs through the code
  - Expected output
  - The functionality of conditional loops
  - Testing of each statement, object, and function on an individual basis

TEST SUITES DESIGN

5.3.1 Test Cases

Table. 5.1 Test Suite 1

Test Suite 01	**Login**						
	Test Case Description	**Test Steps**	**Pre-conditions**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
Test Case 01	To check login functionality	navigate to login page			able to see login dialog box	user logged in	Pass
		enter Email	valid email	namrata@gmail.com	credentials can be entered		
		enter passsword	valid password	namrata	credentials can be entered		
		click login			user logged in		
Test		navigate to			able to see login	invalid	Pass

Case 02		login page			dialog box	credentials	
		enter email	valid email	namrata@gmail.com	credentials can be entered		
		enter password	invalid password	xyz	credentials can be entered		
		click login			invalid credentials		
Test Case 03		navigate to login page			able to see login dialog box	invalid credentials	Pass
		enter email	invalid email	namr@gmail.com	credentials can be entered		
		enter password	invalid password	zzx	credentials can be entered		
		click login			invalid credentials		
Test Case 04		navigate to login page			able to see login dialog box	invalid credentials	Pass
		enter email	invalid email	namr@gmail.com	credentials can be entered		
		enter password	valid password	namrata	credentials can be entered		
		click login			invalid credentials		

Table 5.2 Test Suite 2

Test Suite 02	**Signup**						
	Test Case Description	**Test Steps**	**Pre-Conditions**	**Test Data**	**Expected Result**	**Actual Result**	**Status**
Test Case 01	To check signup functionality	navigate to signup page			able to see signup dialog box	Data successfully recorded	Pass
		First Name		Namrata	credentials can be entered		
		Last Name		Patel			
		Email	vaild Email	abc@gmail.com			

		Create Password		nam123			
		Confirm Password	Correct Confirmed Password	nam123			
		click Get Started					
Test Case 02		navigate to signup page			able to see signup dialog box		
		First Name					
		Last Name					
		Email	Any field with invalid data	Any invalid data	Invalid credentials	Invalid Credentials	Pass
		Confirm Password					
		Re-enter Password					
		click Get Started					

Table 5.3 Test Suite 3

Test Suite 03	**Test Case Description**	**Test Steps**	**Pre-conditions**	**Test data**	**Expected Result**	**Actual Result**	**Status**
Test Case 01	Check the functionality of register property	Navigate to Register page			Able to see Register dialog box		
		City	Select city from options	Anand	Credentials can be entered	Property successfully added	Pass
		Apartment type	Select apartment type from options	Bungalow			
		BHK Type	Select bhk type from options	3bhk			

		Property Size	Built up area (sq.ft)	1700 sq.ft			
		No. of Floors	Select no. of floors of property	1			
Test Case 02		Navigate to Register page			Able to see Register dialog box		
		City	Select city from options	Any of the field not selected	Credentials can be entered	Property not added	Pass
		Apartment type	Select apartment type from options				
		BHK Type	Select bhk type from options				
		Property Size	Built up area (sq.ft)				
		No. of Floors	Select no. of floors of property				

CHAPTER 6

CONCLUSION AND DISCUSSION

SELF ANALYSIS OF PROJECT VIABILITIES

The project was a real challenge for us. As given the better apps in both the App Store and Play Store. We were successful at creating a User-Friendly and Subtle UI. We were also successful in creating database and linking them internally. We were not able to link the users to their events as we expected. We were able to implement authentication via mobile number or Google sign in. We were able to perform CRUD operations with the database using the least internet and very less latency. Our app was able to store data once it was fetched from the database to access the App offline if needed. We were not able to provide the analytics to the admin about their events and user participation as we expected.

PROBLEM ENCOUNTERED AND POSSIBOLITIES SOLUNTIONS

Our major problem that Encountered was to integrate the project, to combine the project together and give a suitable output. We faced many errors while combining the project, Possible Solution that we found during integrating was that to give common IDs for the objects, also we worked together to figure out instant errors and came with certain outputs.

SUMMARY OF PROJECT WORK

Many students, employees and other professionals find difficulties in finding house on rent. Also there are property owners who are searching for a platform to post their property. We investigated such problems brought up the idea for our project. This is the best platform for of them. Also it is such system which excludes the work of middlemen, i.e. the broker and thus, brings transparency in the process.

CHAPTER 7

LIMITATIONS AND FUTURE ENHANCEMENT

LIMITATIONS

- In the low Network zone it will be difficult for the user to locate the Mechanics.
- This application will only work on Android Mobiles.
- It is not cent percent safe from professional hackers.
- Internet is required.

FUTURE ENHANCEMENT

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancement that can be done to this systems are:

- Many more new form of module would be added in the next version of program.
- Based on future security issues, security can be improved using emerging technologies.

REFERENCES

1. https://www.udacity.com/course/firebase-in-a-weekend-by-google-android--ud0352
2. https://www.ebookfrenzy.com/pdf_previews/FirebaseEssentialsAndroidPreview.pdf
3. https://www.youtube.com/watch?v=i-gZAYBMuBs&list=PLGCjwl1RrtcTXrWuRTa59RyRmQ4OedWrt
4. https://www.javatpoint.com/android-tutorial
5. https://developer.android.com/guide