Quinn: Chatbot for Mental Counseling using Rasa Core

A PROJECT REPORT

submitted By

SANDRA V A
TVE16MCA47

to

the APJ Abdul Kalam Technological University
in partial fullfilment of the requirements for the award of the degree

of

Master of Computer Applications

Department of Computer Applications
College of Engineering
Trivandrum-695016

JUNE 2019

DEPARTMENT OF COMPUTER APPLICATIONS

COLLEGE OF ENGINEERING
TRIVANDRUM

CERTIFICATE

This is to certify that the report entitled **Quinn : Chatbot for Mental Counseling using Rasa Core** submitted by **Sandra V A** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications is a bonafide record of the project work carried out by him under my guidance and supervision. This report in any form has not been submitted to any University or Institute for any purpose.

Internal Supervisor External Supervisor

Head of the Dept

ABSTRACT

Many patients undergo psychotherapy due to changes in awareness of psychiatric treatment. Although modern people have many mental illnesses, real rate of the diagnosis and treatment are still low. There is a lack of experts compared to demand and even the cost of consultations is expensive with the experts. In order to solve the problem, a chatbot called "Quinn" is developed to act as a personal assistant for the users. It is a chat robot that interacts with the user to analyze his/her thoughts or state of mind and suggests solutions accordingly. The conversational service can thus provide personalized counseling service to individuals. One-to-one conversation can resolve the isolation effectively. "Quinn" notifies the user's dangerous status when there is an accidental mental disorder, such as panic and suicidal impulse. In addition, the system observes the mood swings continuously for users who have manic-depressive. This conversational service for psychiatric counseling adapts methodologies to understand counseling contents based on of high-level natural language understanding (NLU) using Rasa Core.

ACKNOWLEDGEMENT

If words are considered as symbols of approval and tokens of acknowledgment, then let words play the heralding role of expressing my gratitude.

First of all, I would like to thank God for bestowing me with wisdom, courage and perseverance which had helped me to complete this project ***Quinn- Chatbot for Mental Counseling using Rasa Core***. This project has been a reality as a result of the help given by a large number of personalities.

I would like to remember with gratitude **Prof. Jose T Joseph**, Head Of Department of Computer Applications, College of Engineering, Trivandrum for the encouragement and guidance rendered.

I express my profound and sincere gratitude to **Prof.Vinitha V**, Assistant Professor, Department of Computer Applications, College of Engineering Trivandrum for the immense support and guidance provided by her that aided in the successful completion of our project.

Finally, I wish to express our sincere gratitude to all our friends, who directly or indirectly contributed in this venture.

SANDRA V A

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Now a days more accidents are reported due to over speed and ruthless driving.So this proposed system is used to automate the current bus system. This system uses the concept of line follower. It doesn't require a driver for controlling the bus. This system uses two motors to control rear wheels and two motors for front wheels. It has infrared sensors on the bottom for detection of black tracking line. This sensors detects the black colour and output is given to the comparator.

This system also detect any obstruction or any vehicles that comes in front of bus. This can be achieved using sonar technology. We will be using ultrasonic sensor to detect obstructions. Also it detects bus stop signs using image processing and moves after pre allotted time.

The aim of the study is to provide passengers with comfortable, fast and off-peak public transportation that is not time-consuming in order to minimize the use of private vehicles. With the invention of line following property, vehicles can drive properly. The number of accidents has been reduced thanks to the connection with the host system maintaining information exchange in terms of road conditions and monitoring the vehicle continuously. With the help of the sensors placed on the vehicle, it is possible to store data and exchange information instantly, such as weather and road conditions, general information about the vehicle, the stops, the speed it should follow, the angles and speeds it should have while taking the bends. This study aims to reduce the traffic jam, the time spent on the roads, the pollution and the number of drivers. The newly developed system is based on vehicle – to – infrastructure communication system.

Chapter 2

Requirement Analysis

2.1 Purpose

Mobile robots, is the form of vehicles are able to move along the lines in the roads specially designed as they are taught to follow lines, thus creating a continuous flow in public transport traffic as the robots cannot go out of the roads assigned to them. These robots will be in touch with the host system using the communication modules sending instant data for road condition and the vehicle's current state. Hence, the operator at the host will be able to access the information about the vehicle's current state, temperatures, moisture, speed, location, curves, stops and whether there are objects in front of the vehicle and the distance between the vehicle and the object in meter. The data sent from the vehicle will be analyzed by the operator and used to prevent the possible dangers. The vehicle having the information of a coming curve will slow down to the speed defined before and then accelerate to the speed it had before. This study is important as it has the potential to prevent the accidents arising from the drivers and pedestrians. For example, drivers working in public transport can sometimes drowse off due to the busy and monotonous working hours. Pedestrians crossing the roads carelessly and various objects standing on the roads going unnoticed by the drivers can result in accidents involving death. This study contributes to reducing the potential of risks on the roads by detecting them beforehand.

Numerous studies have been conducted since the word "robot" was coined2. Line follower robots3 have been used in industrial logistics4. Low cost line following system5 and leader and slave robots were designed to do more than one heavy-duty6. On the other hand, robots with digital cameras have been used in mines7, while probes with ultrasonic sensors controlled with fuzzy logic are used in research and rescue operations8, and educational purpose9 as well. General purpose robots development are still being conducted

2.2 Overall Description

This system structure is computationally efficient and can run on a real-time basis. It intends to replace the manual process of driving a bus, uses line follower to move the bus, and image processing for detect the bus stop and after the allotted time bus will move. It helps to reduce the chances of accidents that are caused due to recklessness and overspeed .

Functions

The main functions of the proposed system includes:

- it doesn't require a human intervention
- it reduces accidents caused due to recklessness and over speed
- since it follows the basic rues with which it is programmed with,it will abide by the rules and thus eradicate the cause of traffic congestion

Operating Environment

The operating environment required are:

- **Hardware Requirements**
  Acrylic sheet for body
  4 motor for control movement
  Battery
  Arduino is the main motherboard

- **Software Requirements**
  Language : Python

2.3 Functional requirements

Functional requirements represent the intended behavior of the system. This behavior may be expressed as services, tasks or functions that the specified system is required to perform. The following functional requirements have been identified for this project.

The proposed system consists of 5 modules. They are given below:

- **Automated movement of bus:**

  This module involves building a basic prototype vehicle. For implementing the automated bus system.

  Acrylic sheet for body
  4 motor for control movement
  Battery
  Arduino is the main motherboard
  .

- **Bus line follower:**

  Developed countries like US have specific line for driving bus. This module detects and make sure it stays inside the line. This system involves the use of IR sensor set, complex mathematical functions like Fourier transformation and other mathematical functions.

- **Safety module:**

  This module is used to detect any obstruction or any vehicles that comes in front of bus. This can be achieved using sonar technology. We will be using ultrasonic sensor to detect obstructions.

- **Stop sign:**

  This module involves detecting bus stop signs using image processing. It moves after pre allotted time.

- **Integration:**

  It involves integrating image processing node with vehicle. This can be achieved by using raspberry pi and ZERO W and pi camera.

2.4 Non Functional requirements

Non-Functional requirements define the general qualities of the software product. Non-functional requirement is in effect a constraint placed on the system or the development process. They are usually associated with the product descriptions such as maintainability, usability, portability, etc. it mainly limits the solutions for the problem. The solution should be good enough to meet the non-functional requirements.

Performance Requirements

- Accuracy: Accuracy in functioning and the nature of user-friendliness should be maintained in the system.
- Speed: The system must be capable of offering speed.

Quality Requirements

- Scalability: The software will meet all of the functional requirements.
- Maintainability: The system should be maintainable. It should keep backups to atone for system failures, and should log its activities periodically.
- Reliability: The acceptable threshold for down-time should be long as possible. i.e.mean time between failures should be large as possible. And if the system is broken, time required to get the system back up again should be minimum.

Chapter 3

Design And Implementation

3.1 Overall Design

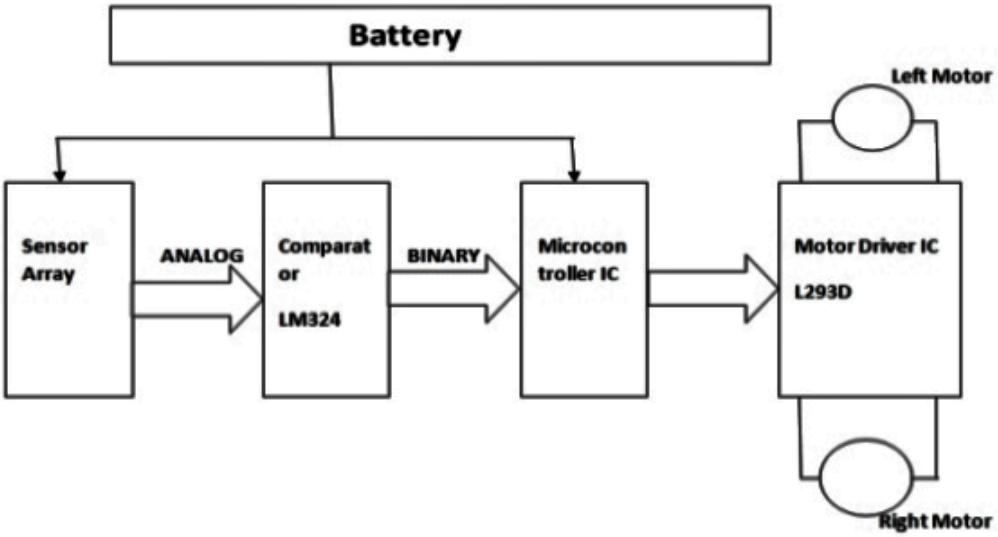

Figure 3.1: system design

3.1.1 System Design

The DFD and the UI design of the application include:

DataFlow Diagram A data flow diagram (DFD) is a design tool to represent the flow of data through an information system. A context level DFD can be used to show the interaction between a system and outside entities; it can also show the internal data flows within a system. It often shows the information system as a single circular shape with no details of its inner working: what it shows is its relationships with the external entities. A data flow diagram graphically represents:

- Processes - jobs that are done with the data. A process transforms incoming data flow into outgoing data flow.
- Data stores - files, databases, archives. They can be manual, digital or temporary.
- External entities - other systems or people beyond the control of the current system.
- Connecting data flows - arrows show how data flows from one place to another.

Notations in a Data Flow Diagram

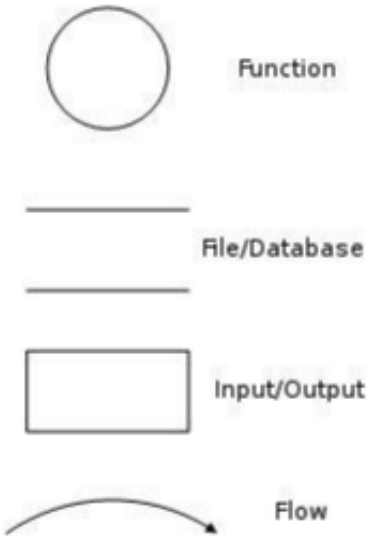

Figure 3.2: Notations in dataflow diagram

Context Diagram (Level 0)

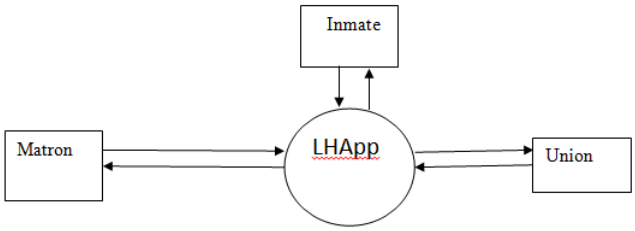

Figure 3.3: Level 0 DFD

Top Level DFD(Level 1)

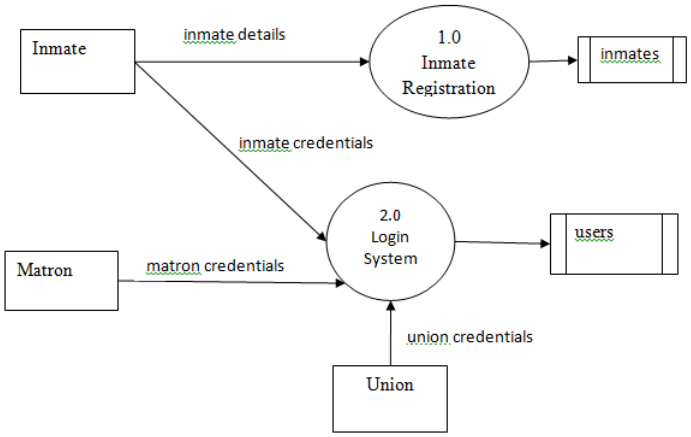

Figure 3.4: Level 1 DFD

Level 2

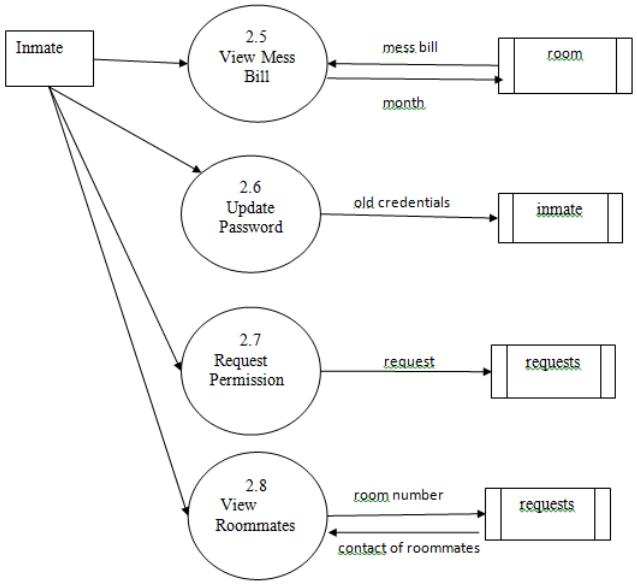

Figure 3.5: Level 2 DFD

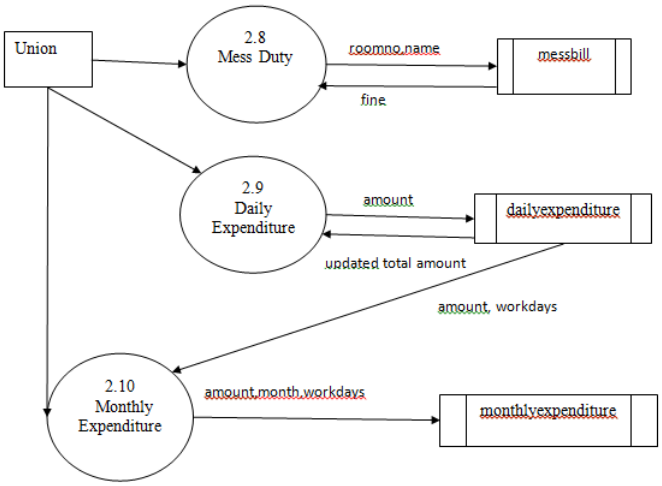

Figure 3.6: Level 2 DFD

Level 3

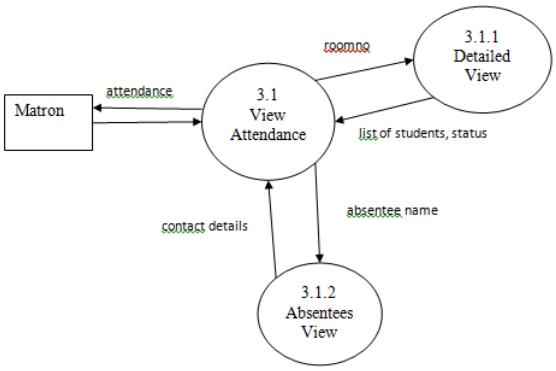

Figure 3.7: Level 3.1 DFD

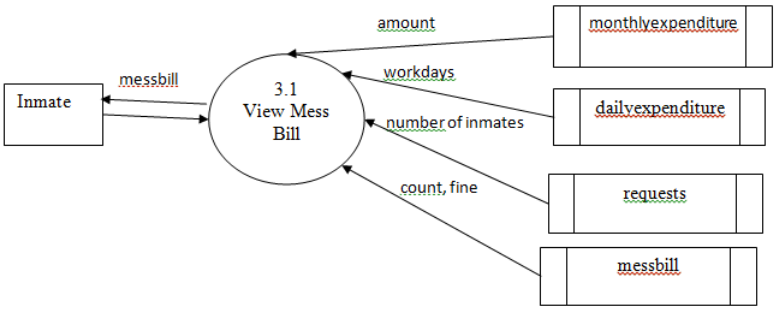

Figure 3.8: Level 3.2 DFD

3.1.2 Database Design

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. The company provides client libraries that enable integration with Android, iOS, JavaScript, Java, Objective-C, Swift and Node.js applications. The database is also accessible through a REST API and bindings for several JavaScript frameworks such as AngularJS, React, Ember.js and Backbone.js. The REST API uses the Server-Sent Events protocol, which is an API for creating HTTP connections for receiving push notifications from a server. Developers using the realtime database can secure their data by using the company's server-side-enforced security rules. Cloud Firestore which is Firebase's next generation of the Realtime Database was released for beta use.

It stores data as key-value pairs. It's a no-sql database and hence is accessed using the child reference with the help of listeners.

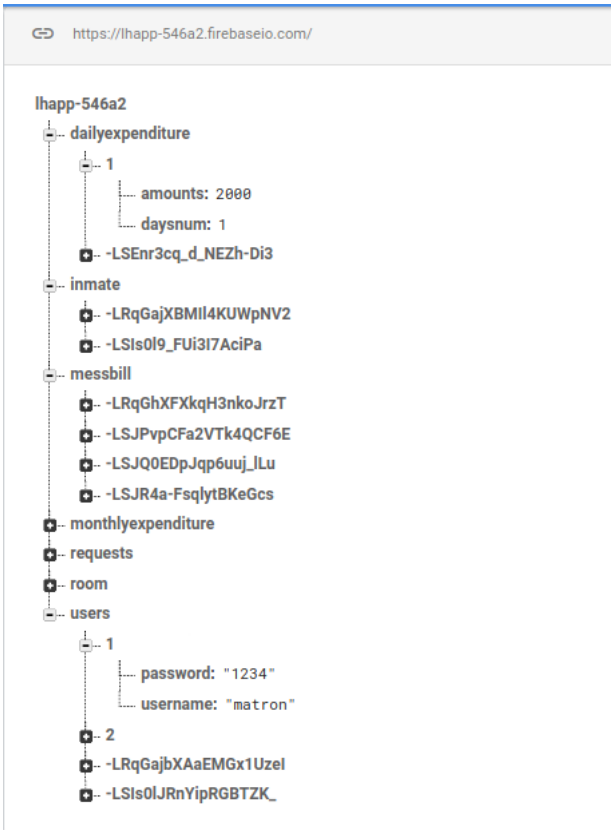

Figure 3.9: Firebase Database Design

3.1.3 User Interface Design

Figure 3.10: Splash screen for Android Application

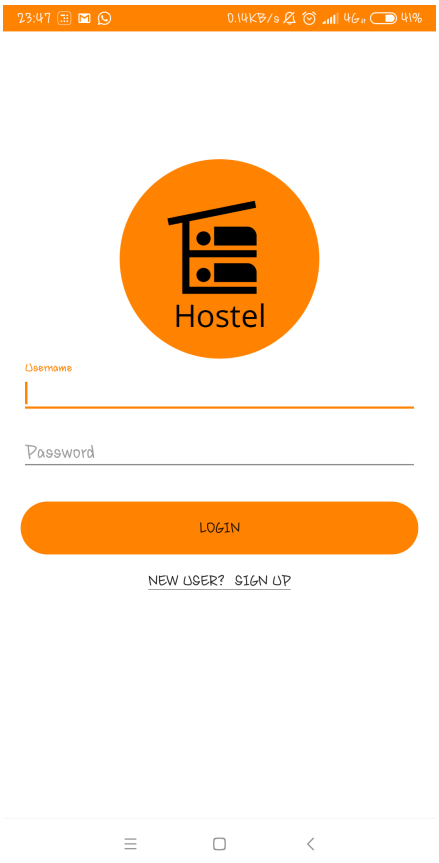

Figure 3.11: Login page

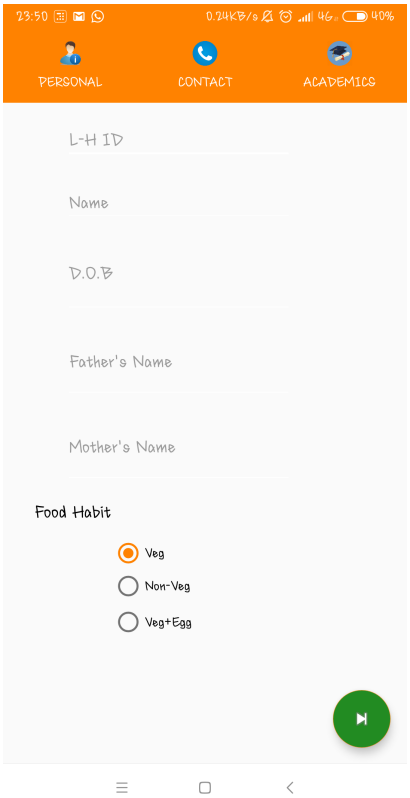

Figure 3.12: User Registration-Fragment 1

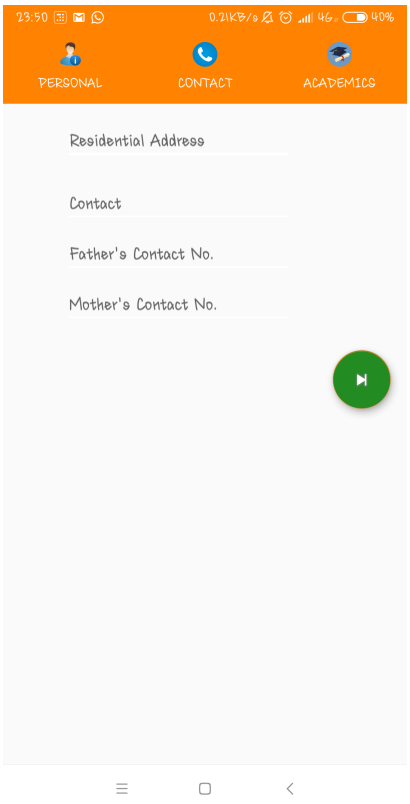

Figure 3.13: User Registration-Fragment 2

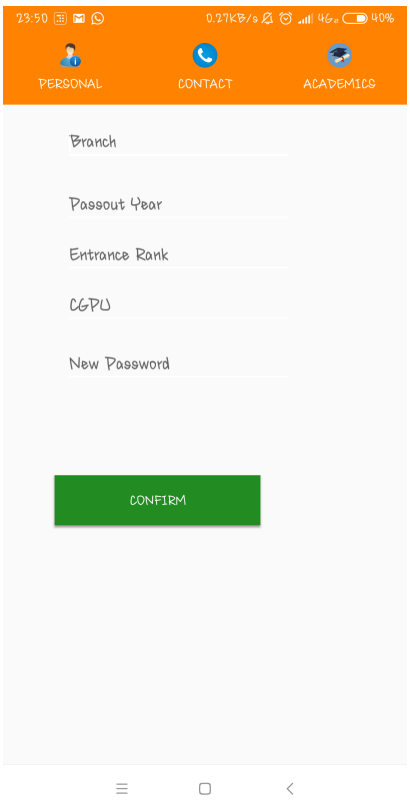

Figure 3.14: User Registration-Fragment 3

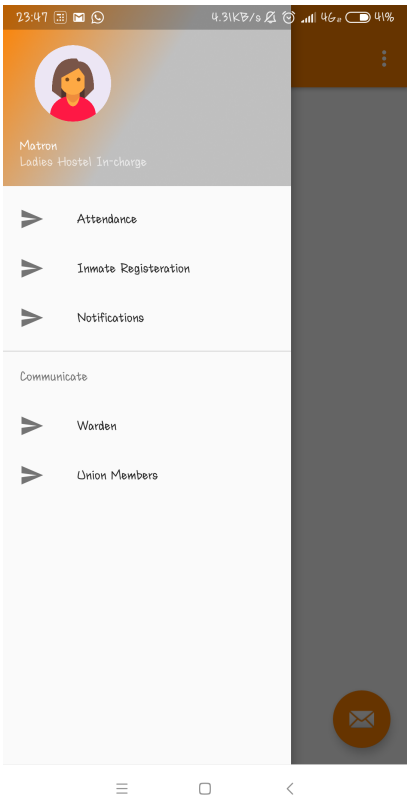

Figure 3.15: Matron Login

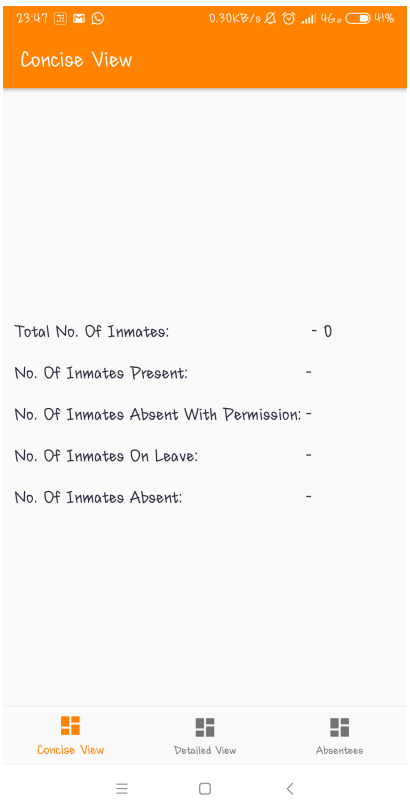

Figure 3.16: Concise Attendance View

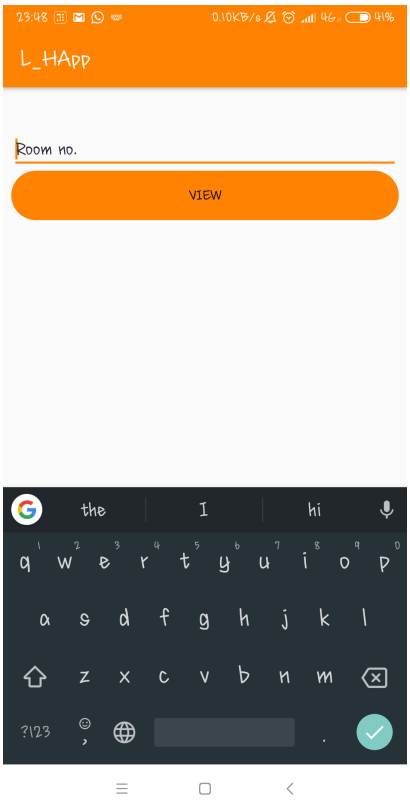

Figure 3.17: Detailed Attendance View

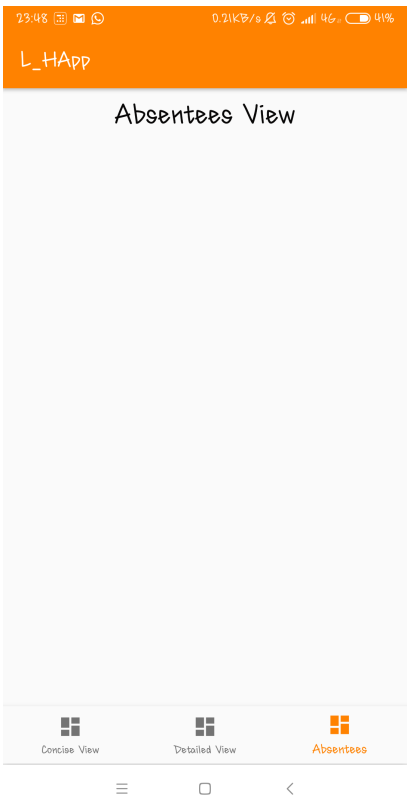

Figure 3.18: Absentees View

Figure 3.19: Registration Confirmation

Figure 3.20: Notification for Successful Room Allotment

Figure 3.21: Contact warden

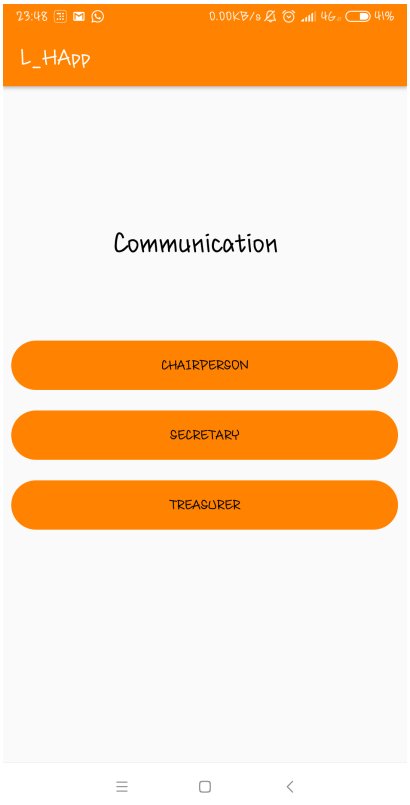

Figure 3.22: Contact union

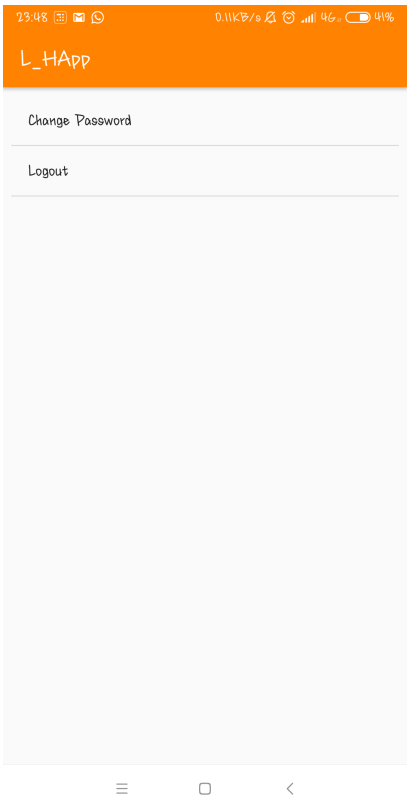

Figure 3.23: Settings

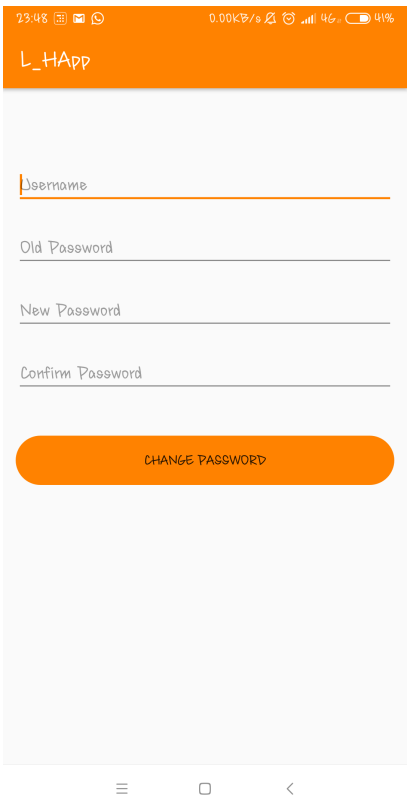

Figure 3.24: Change Password

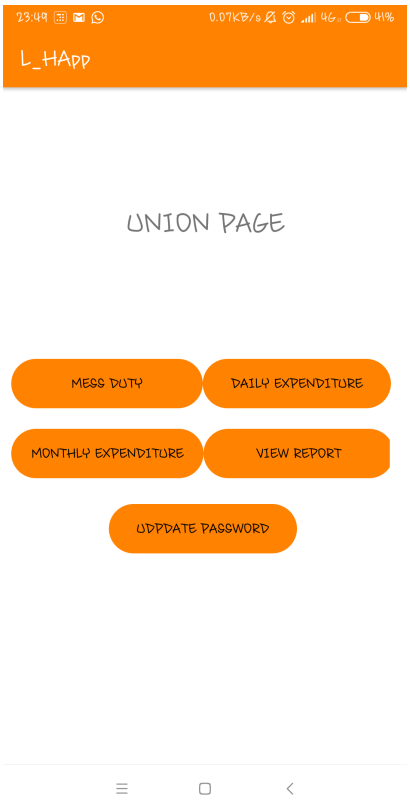

Figure 3.25: Union Login

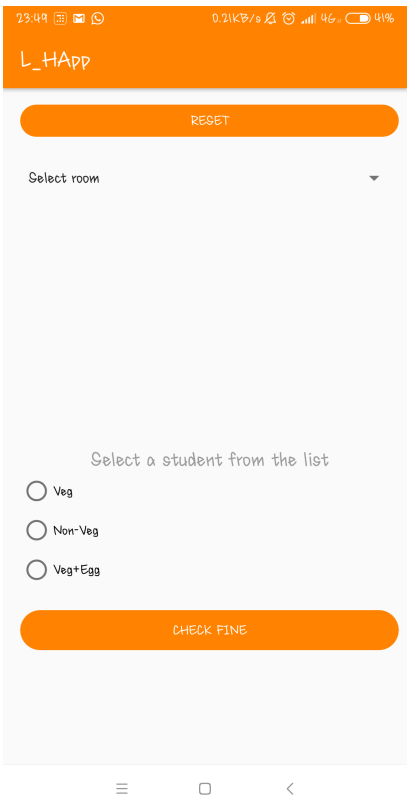

Figure 3.26: Mess Duty

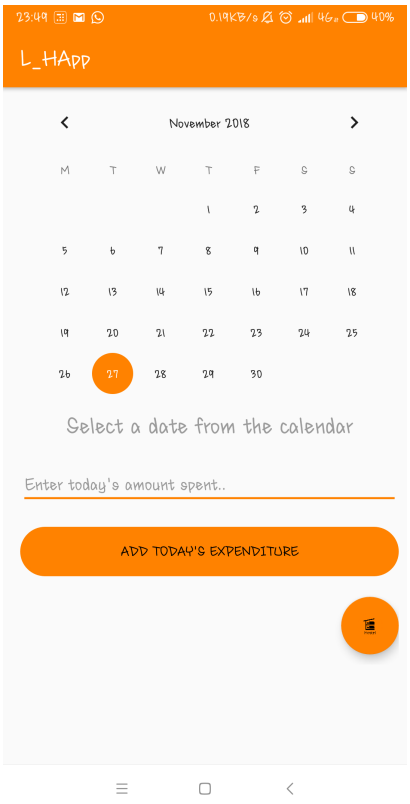

Figure 3.27: Update Daily Expenditure

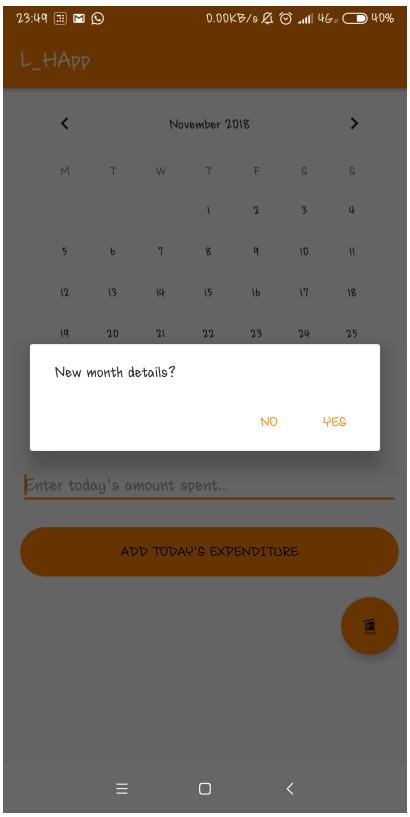

Figure 3.28: Reset all the daily expenditures

Figure 3.29: Update Monthly Expenditure

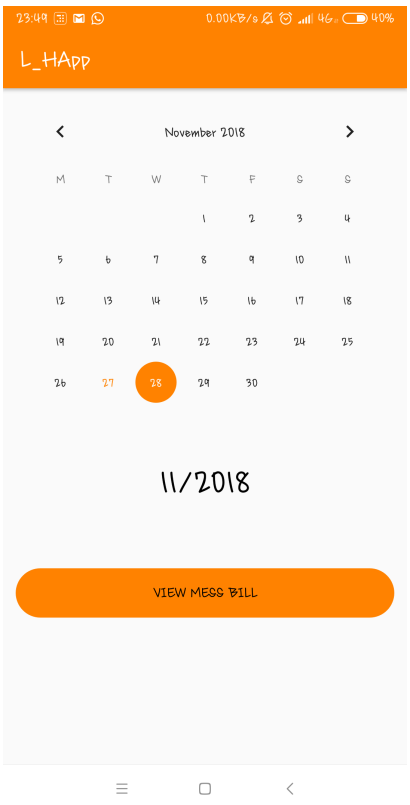

Figure 3.30: View Mess Report

Figure 3.31: Student Login

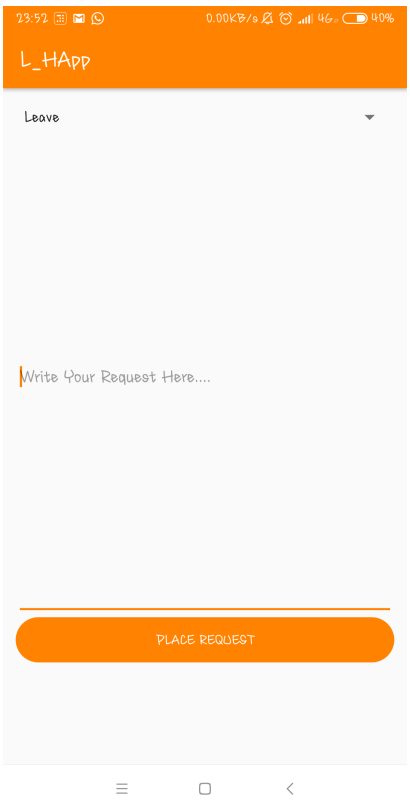

Figure 3.32: Placing requests

Figure 3.33: View Mess Bill of the logged-in user

Chapter 4

Coding

Android Application

Algorithm 1 Algorithm for Android Application is as follows:

1: Create splash screen activity to display welcome screen for 3 second by using Handler class
2: Create activity for login and register. In register button onClickListener launch a new activity with three fragments and fetch user data and store it into the database.
3: In login button onClickListener fetch and compare the data in firebase' real-time database. On success navigate to the homepage of the corresponding user. Create a SharedPreference object and store username and password of user to start session.
4: For matron login : Create a navigation drawer Activity MainActivity with navigation icons as View Attendance, Inmate Registration, Notification, Staff and Union. and create layouts and fragments for each function.
5: For union login : Create an Activity UnionMain with buttons to implement the MessDuty, MonthlyExpenditure, DailyExpenditure, MessReport and UpdatePassword, and create layouts and fragments for each function.
6: For student login : Create an Activity StudentMain with buttons to implement the RoomamteContact, UpdatePassword, PlaceRequests and ViewMessBill, and create layouts and fragments for each function.

Chapter 5

Testing and Implementation

5.1 All the possible testing methods done for the project

System testing is the stage of implementation which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also complete verification to determine whether the objective are met and the user requirements are satisfied.The ultimate aim is quality assurance. Tests are carried and the results are compared with the expected document. In that case of erroneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The test plan defines the unit,integration and system testing approach. The test scope includes the following: A primary objective of testing application systems is to :assure that the system meets the full functional requirements, including quality requirements (Non functional requirements). At the end of the project development cycle, the user should find that the project has met or exceeded all of their expectations as detailed in requirements. Any changes, additions or deletions to the requirements document , functional specification or design specification will be documented and tested at the highest level of quality allowed within the remaining time of the project and within the ability of the test team. The secondary objective of testing application systems will be doing the following: identify and expose all issues and associated risks, communicate all known issues are addressed in an appropriate matter before release This test approach document describes the appropriate strategies, process, work flows and methodologies used to plan, organize, execute and manage testing of software project "LHApp".

5.1.1 Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	Login into the system	Invalid login is Blocked	Same as expected	Pass
2	Register the user	Student can register as an inmate	Registration successful	Pass
3	Contact people	Direct call to the needed authorities	Same as expected	Pass
4	Room allotments	Matron should confirm registration and allot rooms	Registration and room allotment successful	Pass
5	Requests	Students should place requests and matron should respond	Same as expected	Pass
6	View attendance	Matron gets different views of the absentees of the day	Concise,Detailed and Room wise view available	Pass
7	Mess Bill	Compute the monthly mess bill	Same as expected	Pass
8	Update password	Change the password	Same as expected	Pass

Table 5.1: Unit test cases and results

5.1.2 Integration Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	Matron confirms registered user details	Status in inmate db is confirmed and room number is automatically updated in the room db	Same as expected	Pass
2	Room number passed to db	Retrieves inmate list of the given room number	Same as expected	Pass
3	Sends notification if room is confirmed	Inmate receives a confirmation sms	Same as expected	Pass
4	Retrieves requests from db with status="In Progress"	List of inmates who have placed requests	Same as expected	Pass
5	Inmate provides month to view mess bill	Mess for the month is retrieved from the db	Same as expected	Pass

Table 5.2: Integration cases and result-1

Sl No	Procedures	Expected result	Actual result	Pass or Fail
6	Inmate places requests	Request db updated with the given request	Same as expected	Pass
7	Union can compare the inmate menu	Calculate fine if there is a mismatched menu	Same as expected	Pass
8	Union updates daily expenditure and resets at the start of new month	Updates the amount and count in dailyexpenditure db	Same as expected	Pass
9	Selects last day of the month and view the expenditure till now to set the monthly expenditure	Resets the daily expenditure and calculates the amount per head	Same as expected	Pass
10	Selects the month to view the mess bill report	View the monthly mess bill report	Same as expected	Pass

Table 5.3: Integration cases and result-2

5.1.3 System Testing

Text Cases and Result

Sl No	Procedures	Expected result	Actual result	Pass or Fail
1	Generation of mess bill	Generation as per monthly expenditure	Same as expected	Pass
2	Generation of mess report	Depending upon the month	Same as expected	Pass
3	Generation of attendance	Lists the daily attendance	Same as expected	Pass

Table 5.4: System test cases and results

5.2 Advantages and Limitations

The proposed system consists of several advantages compared with previous systems. It automates the daily hectic procedures and other functionalities within the Ladies Hostel. The app manages the hectic task of registering the inmates and their room allotment more effectively. The students can view the monthly mess-bill and also contact their fellow-roommates through this application. The inmates can place requests for permission to the matron who can then accept or reject them. This makes the hostel system more systematic and traceable. The paper work involved in the mess bill calculation of the inmates is reduced using this application, thus removing the delay in generating the mess bill for each month. The manual mess duty is automated which saves effort to a great extend. Daily and monthly expenditures can also be maintained effectively. The generation of monthly mess bill report is another feature offered by this application.

Advantages

- Different views of attendance
- Placing Requests
- Contact authorities or absentees directly

- Sends message when room is allotted
- Manages mess duty
- Maintains daily and monthly expenditure
- View mess bill report
- Database is synced with the app in real-time

There are also some limitations to the proposed system. Large amounts of data in the database may affect the speed of the app. Inmate details once entered cannot be updated later.

5.3 Future Extensions if possible

The proposed system can be extended. This application can be made more effective by linking it with a biometric device to mark the attendance of the inmates. The extended system would then store the fingerprint details of the inmate along with their other details. A provision for updating the registered details of all the users can also be included to make it much more user-friendly.

Chapter 6

Conclusion

The LHApp project is an effective android based application aimed at automating the daily hectic procedures and other functionalities within the Ladies Hostel. The proposed system manages the hectic task of registering the inmates and their room allotment more effectively. The inmates have their individual account where their username is their LHID. The students can view the monthly mess-bill and also contact their fellow-roommates through this application. The inmates can place requests for permission to the matron who can then accept or reject them. This makes the hostel system more systematic and traceable. The paper work involved in the mess bill calculation of the inmates is reduced using this application, thus removing the delay in generating the mess bill for each month. The manual mess duty is automated which saves effort to a great extend. Daily and monthly expenditures can also be maintained effectively. The generation of monthly mess bill report is another feature offered by this application.

.