

Introduction

1.1 Introduction and Objective

Hostel management system is designed to manage all hostel activities like hostel admissions, fees, room, mess allotment, hostel stores & generates related reports for smooth transactions. It is also used to manage monthly mess bill calculation, hostel staff payroll, student certificates, etc.

Designed & developed keeping in mind all the hostel management concerns of staff, Master Soft's Hostel Management Software is a one-stop-solution for higher education institutions. Right from students' registration to allocating rooms & mess, managing student count records, and ins-and-outs of students, it streamlines & automates all day-to-day hostel activities.

Checking the rooms' availability, allocating room to new students, managing hostel fees, visitors' records, mess, and all such hostel management related activities manually consume efforts, time, and resources. Moreover, hostel management also requires looking after the security of students which can only be achieved with close supervision of daily students' activities.

➤ **OBJECTIVE AND SCOPE:**

- To have attractive and Secure Login page to access
- To manage the whole hostel mesh system
- To know what kind of food student and teacher want on daily basis
- Maintain the students as hostellers and waiting list students separately
- Student will no longer apply for their hostels manually
- To know number of items (e.g.: bed, chairs) given to student staying in the hostel
- Students can register their complaints
- Admin can edit notice board and each student can view it
- Hostel secretary can calculate hostel fee including mess fees and can edit mess menu

1.2 Purposed System and It's Advantages

The main objective of the proposed system is to provide information instantly as and when it is required. The main objective is to make the Hostel information details more efficient and effective. This system should maintain different data files, so that the data can be retrieved easily and in an efficient manner. The system is very interactive. It should ensure process integration to the desired extent; various reports should be generated as the need be. This system should also ensure that there is no redundancy in the recorded data.

Advantage:

- Less human error
- Strength and strain of manual labor can be reduced
- High security
- Data redundancy can be avoided to some extent
- Data consistency
- Easy to handle
- Easy data updating
- Easy record keeping
- Backup data can be easily generated
- Efficient usage of resources.
- It provides security through verification process.
- Performance is high.

CHAPTER II

SYSTEM ANALYSIS

2.1 System And Interaction Analysis

The system is user friendly, which doesn't have any complication which is helpful for the teacher and students. System analysis is done when we have analyst the requirement for the development of the application which sets an idea how can we make it happen. System should be high performing with high graphics we are using android studio for the development of the application. Users can interact with the system easily and computerized response will be given to the user if a new query hits, then the staff will reply.

2.2 Analysis

Usage of this application is related to school because it is mainly focused on managing hostel and its mess management it can be helpful for the school with a hostel and huge number of students where there is very difficult to manage students' data manually.

CHAPTER III

FEASIBILITY STUDY

A feasibility study looks at the viability of an idea with an emphasis on identifying potential problems and attempts to answer one main question: Will the idea work and should you proceed with it?

3.1 Technical Feasibility

Technical feasibility refers to the technology used to develop the application. And to find if the product/application meets the organization requirement or not. SO, technical feasibility is also essential in software application development.

3.2 Operational Feasibility

Technical feasibility refers to the application operation and how far the application can meet the organization requirement.

3.3 Economical Feasibility

This is an analysis of the costs to be incurred in the system and the benefits derivable out of the system. Time-based study: This is an analysis of the time required to achieve a return on investments. The future value of a project is also a factor.

JUSTIFICATION AND NEED FOR THE SYSTEM

- The site is built so that users can interact with others.
- To provide better service to their users.
- The time of the user is being saved.
- Allowing the user to contribute to the environment by making less use of paper.

CHAPTER IV

REQUIREMENT SPECIFICATION

4.1 Performance Requirement

Only when the requirement specification is properly given, it is possible to design a system. Which will fit into required environment. It rests largely requirement with the users of the existing system to give the requirement specification because they are the people who finally use the system. This because the requirement has to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

4.2 Hardware Requirement:

- ✓ Screen Resolution-1280 *800
- ✓ CPU-2nd gen intel CPU (Sandy Bridge) or newer, AMD CPU with support for windows Hyper vision.
- ✓ Memory-4GB or 8 GB RAM (Recommended)
- ✓ Free Storage-8GB

4.3 Software Requirements:

- ✓ Operating system-windows 8/8.1/10/11(64 bit)
- ✓ Android Studio with SDK
- ✓ Emulator
- ✓ Java

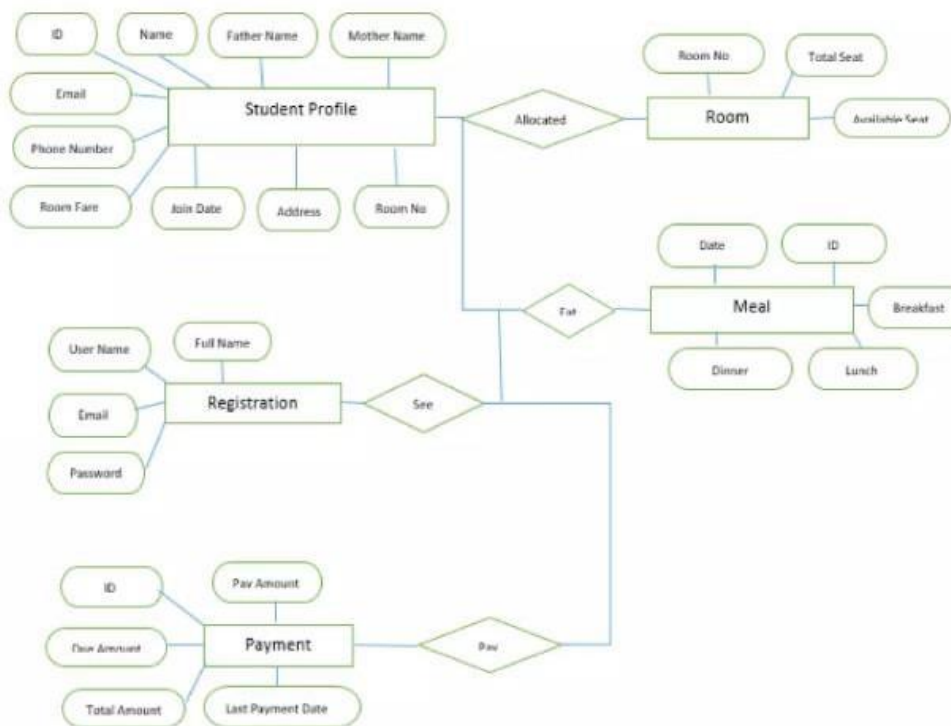
CHAPTER V

SYSTEM DESIGN

5.1 Introduction

System Design is the blueprint or brainstorming need to design or built any software. Software Design means the appearance or the client-side view of the software which a client sees so, it needs to be very unique, attractive and user friendly. So, before developing a software we need to design what kind of software we are going to build. It would be useful for the client or not and how much benefit can client take from our developed application.

5.2 ER-Diagram



CHAPTER VI

SYSTEM TESTING

6.1 Introduction

System testing is the process of finding any error, bugs and vulnerabilities which can cause inconvenience to the client and may cause more time and cost to fix it again after delivering the product to the client. And to find out if the product can match client requirements. System testing is done by the tester even if they are also involved in the development process.

6.2 Security Testing

Security testing is done to check whether there is any loophole in the project which can cause the client to lose their privacy, personal data and other assets which can cause problem in their personal and professional life. Security testing is a important role for software development to ensure the security of client in every way possible.

CHAPTER VII

SYSTEM SECURITY

7.1 Introduction

System security describes the controls and safeguards that an organization takes to ensure its networks and resources are safe from downtime, interference or malicious intrusion.

7.2 Security in Software

In this software we don't have very high security, but we have a simple login page which ask a client/user to verify their identity every single time they visit the software and the password and Gmail/username must match with the previous one in the case that user forgot their previous passwordthey can use forgot password option to reset their password.

CHAPTER VIII

8.1 Layout

activity_absence_notification.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".AbsenceNotification">
    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingBottom="40dp">
        <ImageView
            android:id="@+id/arrow_2"
            android:layout_width="35dp"
            android:layout_height="35dp"
            android:layout_marginStart="25dp"
            android:layout_marginTop="20dp"
            android:src="@drawable/arrow_1"
            tools:ignore="ContentDescription"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintStart_toStartOf="parent"/>
        <TextView
            android:id="@+id/absence_not"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="25dp"
            android:layout_marginTop="30dp"
            android:fontFamily="@font/josefin_sans_bold"
            android:text="@string/absence_not"
            android:textAppearance="@style/absence_not"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/arrow_2"/>
        <TextView
            android:id="@+id/start_date"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="25dp"
            android:layout_marginTop="50dp"
            android:fontFamily="@font/josefin_sans_semibold"
            android:text="@string/start_date"
            android:textAppearance="@style/start_date"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@id/absence_not"/>
        <EditText
            android:id="@+id/start_date_input"
            android:layout_width="match_parent"
            android:inputType="text"
            android:hint="04/10/21"
            android:textColor="#000000"
            android:layout_height="wrap_content"
            android:backgroundTint="#39374D"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintEnd_toEndOf="parent">
```

```

android:layout_marginEnd="25dp"
android:layout_marginStart="25dp"/>
<TextView
android:id="@+id/end_date"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="25dp"
android:layout_marginTop="20dp"
android:fontFamily="@font/josefin_sans_semibold"
android:text="@string/end_date"
android:textAppearance="@style/start_date"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@id/start_date_input"/>
<EditText
android:id="@+id/end_date_input"
android:layout_width="match_parent"
android:inputType="text"
android:hint="07/10/21"
android:textColor="#000000"
android:layout_height="wrap_content"
android:backgroundTint="#39374D"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toBottomOf="@id/end_date"
android:layout_marginEnd="25dp"
android:layout_marginStart="25dp"/>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:layout_marginStart="25dp"
android:layout_marginTop="40dp"
android:layout_marginEnd="25dp"
app:layout_constraintTop_toBottomOf="@id/end_date_input"
android:orientation="horizontal"
android:weightSum="2">
<Button
android:id="@+id/cancel"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:elevation="10dp"
android:layout_marginEnd="20dp"
android:background="@drawable/white_background"
android:fontFamily="@font/josefin_sans_semibold"
android:text="@string/cancel"
android:textAppearance="@style/cancel"
android:textAllCaps="false"
android:layout_weight="1"/>
<Button
android:id="@+id/OK"
android:layout_width="0dp"
android:fontFamily="@font/josefin_sans_semibold"
android:layout_height="wrap_content"
android:elevation="10dp"
android:background="@drawable/white_background"
android:textAppearance="@style/ok"
android:text="@string/ok"
android:layout_weight="1"/>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
</RelativeLayout>

```


activity_balance.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".payment.BalanceActivity">
    <TextView
        android:id="@+id/balance_heading"
        android:layout_width="340dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:text="@string/balance"
        android:layout_marginTop="30dp"
        android:layout_marginStart="20dp"
        android:textAppearance="@style/edit_your_details"
        android:textSize="33sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <ImageView
        android:id="@+id/breakfast_image"
        android:layout_width="340dp"
        android:layout_height="170dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp"
        android:contentDescription="@string/breakfast_image"
        android:src="@drawable/breakfast"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/balance_heading" />
    <TextView
        android:id="@+id/balance_txt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:fontFamily="@font/josefin_sans_bold"
        android:text="@string/your_balance"
        android:textAlignment="center"
        android:textSize="30sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/breakfast_image" />
    <TextView
        android:id="@+id/balance"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:fontFamily="@font/josefin_sans_bold"
        android:text="-----"
        android:layout_marginTop="10dp"
        android:textAlignment="center"
        android:textColor="#000"
        android:textSize="38sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/balance_txt" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_complaint.xml

```

<?xml version="1.0" encoding="utf-8"?>

```

```

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context="in.hackslash.messsy.complaint.ComplaintActivity">
<RelativeLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:paddingBottom="43dp">
<EditText
android:id="@+id/editTextTextPersonName"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignStart="@+id/rectangle_4"
android:layout_alignTop="@+id/rectangle_4"
android:layout_alignEnd="@+id/rectangle_4"
android:layout_marginStart="0dp"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@null"
android:ems="10"
android:scrollbars="vertical"
android:inputType="textMultiLine"
android:maxLines="7" />
<TextView
android:id="@+id/complaint_s"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="25dp"
android:layout_marginTop="30dp"
android:textSize="28sp"
android:textColor="#000"
android:fontFamily="@font/josefin_sans_bold"
android:text="Complaint Section" />
<TextView
android:id="@+id/issue"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/complaint_s"
android:layout_marginStart="25dp"
android:layout_marginTop="52dp"
android:textColor="#000"
android:fontFamily="@font/josefin_sans_bold"
android:text="Issue"/>
<EditText
android:id="@+id/line_1"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/issue"
android:layout_marginStart="25dp"
android:layout_marginEnd="30dp"
android:backgroundTint="#39374D"
android:inputType="text"
android:textColor="#000000"
tools:ignore="Autofill,LabelFor" />

```

```

<TextView
android:id="@+id/description"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/line_1"

```

```

android:layout_marginTop="28dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="Description"
android:textColor="#000"/>
<EditText
android:id="@+id/line_2"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/description"
android:layout_marginStart="25dp"
android:layout_marginEnd="30dp"
android:backgroundTint="#39374D"
android:inputType="text"
android:textColor="#000000"
tools:ignore="Autofill,LabelFor" />

```

```

<TextView
android:id="@+id/img"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/line_2"
android:layout_marginStart="25dp"
android:layout_marginTop="28dp"
android:textColor="#000"
android:fontFamily="@font/josefin_sans_bold"
android:text="Complaint Image" />

```

```

<ImageView
android:id="@+id/rectangle_4"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/img"
android:layout_centerHorizontal="true"
android:layout_marginStart="25dp"
android:layout_marginEnd="30dp"
android:scaleType="fitXY"
android:src="@drawable/description_box"
tools:ignore="ContentDescription" />

```

```

<Button
android:id="@+id/upload_button"
android:layout_width="135dp"
android:layout_height="38dp"
android:layout_below="@id/rectangle_4"
android:layout_marginStart="25dp"
android:layout_marginTop="24dp"
android:background="@drawable/white_background"
android:elevation="12dp"
android:fontFamily="@font/josefin_sans_bold"
android:gravity="center"
android:text="Upload Image"
android:textColor="#000"/>

```

```

<Button
android:id="@+id/file_complaint_button1"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/upload_button"
android:layout_marginStart="35dp"
android:layout_marginTop="80dp"
android:layout_marginEnd="35dp"
android:background="@drawable/rectangle_1"

```

```

android:fontFamily="@font/josefin_sans_bold"
android:text="File Complaint"
android:textColor="#fff" />
</RelativeLayout>
</ScrollView>

```

activity_create_account.xml

```

<?xml version="1.0" encoding="utf-8"?>

<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".onboarding.CreateAccountActivity"
    app:layout_constraintStart_toStartOf="parent">

    <RelativeLayout
        android:id="@+id/create_account"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#FFFFFF"
        android:orientation="vertical"
        tools:context=".MainActivity">

        <ProgressBar
            android:id="@+id/signup_progress"
            style="@style/Widget.AppCompat.ProgressBar.Horizontal"
            android:layout_width="match_parent"
            android:visibility="invisible"
            android:layout_height="wrap_content"
            android:indeterminate="true" />

        <TextView
            android:id="@+id/welcome_to_"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentTop="true"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="25dp"
            android:text="@string/welcome_to_"
            android:textAppearance="@style/welcome_to_"
            android:textSize="35sp" />

        <ImageView
            android:id="@+id/breakfast_image"
            android:layout_width="340dp"
            android:layout_height="160dp"
            android:layout_alignParentTop="true"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="120dp"
            android:src="@drawable/breakfast" />

        <TextView
            android:id="@+id/name"
            android:layout_width="wrap_content"

```

```

android:layout_below="@id/breakfast_image"
android:layout_alignStart="@id/name_input"
android:layout_marginStart="3dp"
android:layout_marginTop="28dp"
android:text="@string/name"
android:textAppearance="@style/name" />

```

```

<EditText
android:id="@+id/name_input"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/name"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="textPersonName"
android:textColor="#000000" />

```

```

<TextView
android:id="@+id/enter_roll"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/name_input"
android:layout_alignStart="@id/password_input"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="Roll Number"
android:textAppearance="@style/enter_password" />

```

```

<EditText
android:id="@+id/input_roll"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/enter_roll"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="number"
android:textColor="#000000" />

```

```

<TextView
android:id="@+id/email_address"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/input_roll"
android:layout_alignStart="@id/email_input"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/email_add"
android:textAppearance="@style/email_add" />

```

```

<EditText
android:id="@+id/email_input"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/email_address"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"

```

```
android:textColor="#000000" />
```

```
<TextView
android:id="@+id/room_no"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/email_input"
android:layout_alignStart="@id/room_no_input"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/room_no"
android:textAppearance="@style/room_no" />
```

```
<EditText
android:id="@+id/room_no_input"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/room_no"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="text|number"
android:textColor="#000000" />
```

```
<TextView
android:id="@+id/enter_password"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/room_no_input"
android:layout_alignStart="@id/password_input"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/enter_password"
android:textAppearance="@style/enter_password" />
```

```
<EditText
android:id="@+id/password_input"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/enter_password"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="textPassword"
android:textColor="#000000" />
```

```
<Button
android:id="@+id/create_acc"
android:layout_width="290dp"
android:layout_height="55dp"
android:layout_below="@id/password_input"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="40dp"
android:background="@drawable/rectangle_1"
android:elevation="10dp"
android:text="@string/create_account"
android:textColor="#FFFFFF"
android:textSize="18sp" />
```

```

android:id="@+id/already_have_an_account"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/create_acc"
android:layout_alignStart="@id/create_acc"
android:layout_marginStart="65dp"
android:layout_marginTop="9dp"
android:layout_marginBottom="20dp"
android:text="@string/already_have_an_account"
android:textAppearance="@style/already_have_an_account" />

```

```

<TextView
android:id="@+id/log_in"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/create_acc"
android:layout_alignBaseline="@id/already_have_an_account"
android:layout_alignStart="@id/create_acc"
android:layout_marginStart="212dp"
android:layout_marginBottom="20dp"
android:text="@string/log_in"
android:textAppearance="@style/log_in" />

```

```

</RelativeLayout>
</ScrollView>

```

activity_edit_profile.xml

```

<?xml version="1.0" encoding="utf-8"?>

```

```

<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".onboarding.CreateAccountActivity">

```

```

<RelativeLayout
android:id="@+id/edit_profile"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:background="#FFFFFF"
android:orientation="vertical"
tools:context=".MainActivity">

```

```

<TextView
android:id="@+id/edit_your_details"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_centerHorizontal="true"
android:layout_alignParentTop="true"
android:layout_marginTop="30dp"
android:text="@string/edit_your_details"
android:textAppearance="@style/edit_your_details"
android:textSize="33sp"/>

```

```

<androidx.constraintlayout.widget.ConstraintLayout
android:id="@+id/cl"

```

```

android:layout_height="200dp"
android:layout_marginTop="120dp">
<ImageView
android:id="@+id/profile_img"
android:layout_width="150dp"
android:layout_height="150dp"
android:src="@drawable/undraw_profile_pic"
android:scaleType="centerCrop"
android:background="@drawable/white_background"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toTopOf="parent" />
<ImageButton
android:id="@+id/profile_img_upload_button"
android:layout_width="25dp"
android:layout_height="25dp"
android:background="@drawable/white_background"
android:backgroundTint="@color/blue"
android:src="@drawable/ic_round_add"
app:layout_constraintEnd_toEndOf="@id/profile_img"
app:layout_constraintTop_toTopOf="@id/profile_img"
android:layout_marginEnd="16dp"
android:layout_marginTop="10dp"/>

<Button
android:id="@+id/profile_img_set_button"
android:visibility="invisible"
android:layout_width="180dp"
android:layout_height="38dp"
android:background="@drawable/white_background"
android:elevation="12dp"
android:fontFamily="@font/josefin_sans_bold"
android:gravity="center"
android:text="Set As Profile Image"
android:textColor="#000"
android:layout_marginTop="20dp"
app:layout_constraintTop_toBottomOf="@id/profile_img"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintBottom_toBottomOf="parent"
android:layout_marginBottom="20dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>

```

```

<TextView
android:id="@+id/name_text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/cl"
android:layout_marginTop="28dp"
android:layout_alignStart="@id/name"
android:layout_marginStart="3dp"
android:text="@string/name"
android:textAppearance="@style/name" />

```

```

<EditText
android:id="@+id/name"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/name_text"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"

```



```

android:backgroundTint="#39374D"
android:inputType="text"
android:textColor="#000000"
android:autofillHints="" />

```

```

<TextView
android:id="@+id/roll_text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/name"
android:layout_alignStart="@id/email"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="Roll Number"
android:textAppearance="@style/email" />

```

```

<EditText
android:id="@+id/roll"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/roll_text"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="text"
android:textColor="#000000"
android:autofillHints="" />

```

```

<TextView
android:id="@+id/email_address_text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/roll"
android:layout_alignStart="@id/email"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/email_add"
android:textAppearance="@style/email" />

```

```

<EditText
android:id="@+id/email"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/email_address_text"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="textEmailAddress"
android:textColor="#000000"
android:autofillHints="" />

```

```

<TextView
android:id="@+id/room_no_text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/email"
android:layout_alignStart="@id/room_no"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/room_no"

```

```

<EditText
    android:id="@+id/room_no"
    android:layout_width="340dp"
    android:layout_height="wrap_content"
    android:layout_below="@id/room_no_text"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="40dp"
    android:layout_marginTop="0dp"
    android:backgroundTint="#39374D"
    android:inputType="text"
    android:textColor="#000000"
    android:autofillHints="" />

```

```

<TextView
    android:id="@+id/enter_password_text"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/room_no"
    android:layout_alignStart="@id/password"
    android:layout_marginStart="3dp"
    android:layout_marginTop="0dp"
    android:text="@string/enter_password"
    android:textAppearance="@style/password" />

```

```

<EditText
    android:id="@+id/password"
    android:layout_width="340dp"
    android:layout_height="wrap_content"
    android:layout_below="@id/enter_password_text"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="40dp"
    android:layout_marginTop="0dp"
    android:backgroundTint="#39374D"
    android:inputType="textPassword"
    android:textColor="#000000"
    android:importantForAutofill="no" />

```

```

<Button
    android:id="@+id/save_button"
    android:layout_width="290dp"
    android:layout_height="55dp"
    android:layout_below="@id/password"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="32dp"
    android:layout_marginBottom="40dp"
    android:background="@drawable/rectangle_1"
    android:elevation="10dp"
    android:text="@string/save_details"
    android:textAllCaps="false"
    android:textColor="#FFFFFF"
    android:textSize="18sp" />

```

```

<ProgressBar
    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:visibility="invisible"
    android:indeterminate="true"/>

```

```

</RelativeLayout>

```

</ScrollView>

activity_final_payment\

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".payment.FinalPaymentActivity">
```

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="64dp"
    android:src="@drawable/breakfast"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="24dp"
    android:gravity="center"
    android:text="Total Payable"
    android:textSize="25sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="@+id/imageView"
    app:layout_constraintStart_toStartOf="@+id/imageView"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

```
<TextView
    android:id="@+id/amount_final"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="84dp"
    android:gravity="center"
    android:text="25Rs"
    android:textColor="@color/black"
    android:textSize="30sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="@+id/imageView"
    app:layout_constraintHorizontal_bias="0.506"
    app:layout_constraintStart_toStartOf="@+id/imageView"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

```
<EditText
    android:id="@+id/payment_final"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="49dp"
    android:ems="7"
    android:gravity="center"
    android:hint="Enter roll number"
    android:inputType="number"
    app:layout_constraintEnd_toEndOf="@+id/amount_final"
    app:layout_constraintStart_toStartOf="@+id/amount_final"
```

```

<Button
    android:id="@+id/final_pay"
    android:layout_width="172dp"
    android:layout_height="58dp"
    android:layout_marginTop="68dp"
    android:background="@drawable/curved_rectangle"
    android:text="pay"
    android:textColor="@color/profileBackground"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="@+id/payment_final"
    app:layout_constraintHorizontal_bias="0.483"
    app:layout_constraintStart_toStartOf="@+id/payment_final"
    app:layout_constraintTop_toBottomOf="@+id/payment_final" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_forgot_password.xml

```

<?xml version="1.0" encoding="utf-8"?>

<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".onboarding.ForgotPassword"
    app:layout_constraintStart_toStartOf="parent">

    <RelativeLayout
        android:id="@+id/forgot_password"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="#FFFFFF"
        android:orientation="vertical">

        <TextView
            android:id="@+id/forgot_pass"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentTop="true"
            android:layout_alignStart="@id/email_text"
            android:layout_marginTop="25dp"
            android:text="@string/forgot_password"
            android:textAppearance="@style/forgot_pass"
            android:textSize="35sp" />

        <ImageView
            android:id="@+id/breakfast_image"
            android:layout_width="340dp"
            android:layout_height="170dp"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="120dp"
            android:src="@drawable/breakfast"
            android:contentDescription="@string/breakfast_image" />

        <TextView
            android:id="@+id/email_text"
            android:layout_width="wrap_content"

```

```

android:layout_height="wrap_content"
android:layout_below="@id/breakfast_image"
android:layout_alignStart="@id/email_edit_text"
android:layout_marginTop="28dp"
android:layout_marginStart="3dp"
android:text="@string/email"
android:textAppearance="@style/name" />

```

```

<EditText
android:id="@+id/email_edit_text"
android:layout_width="340dp"
android:layout_height="wrap_content"
android:layout_below="@id/email_text"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="0dp"
android:backgroundTint="#39374D"
android:inputType="textEmailAddress"
android:textColor="#000000" />

```

```

<Button
android:id="@+id/btn_forgot_pass"
android:layout_width="290dp"
android:layout_height="55dp"
android:layout_below="@id/email_edit_text"
android:layout_centerHorizontal="true"
android:layout_marginStart="40dp"
android:layout_marginTop="220dp"
android:background="@drawable/rectangle_1"
android:elevation="10dp"
android:text="@string/get_password_reset_link"
android:textColor="#FFFFFF"
android:textSize="18sp" />

```

```

</RelativeLayout>
</ScrollView>

```

activity_home.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer_layout"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fitsSystemWindows="true"
tools:context=".home.HomeActivity"
tools:openDrawer="end">

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".home.HomeActivity">

<androidx.appcompat.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="?attr/actionBarSize"

```

```

android:elevation="4dp"
app:popupTheme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="#f9db22"
    android:textSize="20sp" />
</androidx.appcompat.widget.Toolbar>

```

```

<!-- <TextView-->
<!--     android:id="@+id/food_name_tv"-->
<!--     android:layout_width="match_parent"-->
<!--     android:layout_height="wrap_content"-->
<!--     android:layout_below="@id/delight_tv"-->
<!--     android:layout_marginStart="22dp"-->
<!--     android:layout_marginTop="4dp"-->
<!--     android:text="Panner Butter Masala"-->
<!--     android:textColor="#000000"-->
<!--     android:textSize="16sp"-->
<!--     android:textStyle="bold" />-->

```

```

<!-- <TextView-->
<!--     android:id="@+id/food_intro_tv"-->
<!--     android:layout_width="match_parent"-->
<!--     android:layout_height="wrap_content"-->
<!--     android:layout_below="@id/food_name_tv"-->
<!--     android:layout_marginStart="22dp"-->
<!--     android:layout_marginTop="4dp"-->
<!--     android:text="Detail of food Item"-->
<!--     android:textColor="#000000"-->
<!--     android:textSize="13sp" />-->

```

```

<!-- </RelativeLayout>-->
<FrameLayout
    android:id="@+id/frameContainer"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_above="@+id/bottom_layout"
    android:layout_below="@+id/toolbar" />

```

```

<androidx.coordinatorlayout.widget.CoordinatorLayout
    android:id="@+id/bottom_layout"
    android:layout_width="match_parent"
    android:layout_height="80dp"
    android:layout_alignParentBottom="true"
    android:theme="@style/NavTheme">

```

```

<com.google.android.material.bottomappbar.BottomAppBar
    android:id="@+id/bottomAppBar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:fabCradleMargin="8dp"
    android:backgroundTint="@color/main_colour"
    app:fabCradleRoundedCornerRadius="20dp"
    android:layout_gravity="bottom">

```

```

<com.google.android.material.bottomnavigation.BottomNavigationView
    android:id="@+id/btm_nav"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```

android:layout_alignParentBottom="true"
android:background="@android:color/transparent"
app:itemIconTint="@drawable/menu_color"
app:itemTextColor="@drawable/menu_color"
app:menu="@menu/menu"/>

</com.google.android.material.bottomappbar.BottomAppBar>

<!--      <com.google.android.material.floatingactionbutton.FloatingActionButton-->
<!--      android:id="@+id/fab"-->
<!--      android:layout_width="wrap_content"-->
<!--      android:layout_height="wrap_content"-->
<!--      android:src="@drawable/scan"-->
<!--      app:layout_anchor="@id/bottomAppBar" />-->
</androidx.coordinatorlayout.widget.CoordinatorLayout>
<!--      <com.google.android.material.bottomnavigation.BottomNavigationView-->
<!--      android:id="@+id/btm_nav"-->
<!--      android:layout_width="match_parent"-->
<!--      android:layout_height="wrap_content"-->
<!--      android:layout_alignParentBottom="true"-->
<!--      app:itemIconTint="@drawable/menu_color"-->
<!--      app:itemTextColor="@drawable/menu_color"-->
<!--      app:menu="@menu/menu" />-->

</RelativeLayout>
<com.google.android.material.navigation.NavigationView
android:id="@+id/nav_view"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:layout_gravity="start"
app:headerLayout="@layout/nav_header"
app:menu="@menu/nav_menu"
android:background="#373549"
app:itemTextColor="#fff"
/>

</androidx.drawerlayout.widget.DrawerLayout>

```

activity_login.xml

```

<?xml version="1.0" encoding="utf-8"?>

<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".onboarding.LoginActivity">

<RelativeLayout
android:id="@+id/sign_in"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:background="#FFFFFF"
tools:context=".MainActivity">

<ProgressBar

```

```

style="@style/Widget.AppCompat.ProgressBar.Horizontal"
android:layout_width="match_parent"
android:visibility="invisible"
android:layout_height="wrap_content"
android:indeterminate="true" />

```

```

<ImageView
android:id="@+id/img_1"
android:layout_width="wrap_content"
android:layout_height="320dp"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginStart="40dp"
android:layout_marginTop="10dp"
android:layout_marginEnd="50dp"
android:src="@drawable/logo"
tools:ignore="ContentDescription" />

```

```

<TextView
android:id="@+id/welcome_to_"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/img_1"
android:layout_centerHorizontal="true"
android:layout_marginTop="12dp"
android:text="@string/welcome_to_"
android:textAppearance="@style/welcome_to_"
android:textSize="32sp" />

```

```

<TextView
android:id="@+id/email"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/welcome_to_"
android:layout_alignStart="@id/line_1"
android:layout_marginStart="3dp"
android:layout_marginTop="16dp"
android:text="@string/email"
android:textAppearance="@style/email" />

```

```

<EditText
android:id="@+id/line_1"
android:layout_width="320dp"
android:layout_height="wrap_content"
android:layout_below="@id/email"
android:layout_centerHorizontal="true"
android:layout_marginTop="8dp"
android:backgroundTint="#39374D"
android:inputType="textEmailAddress"
android:textColor="#000000" />

```

```

<TextView
android:id="@+id/password"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/line_1"
android:layout_alignStart="@id/line_1"
android:layout_marginStart="3dp"
android:layout_marginTop="0dp"
android:text="@string/Password"
android:textAppearance="@style/password"

```



```

<EditText
    android:id="@+id/line_2"
    android:layout_width="320dp"
    android:layout_height="wrap_content"
    android:layout_below="@id/password"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="8dp"
    android:backgroundTint="#39374D"
    android:inputType="textPassword"
    android:textColor="#000000" />

<Button
    android:id="@+id/rectangle_1"
    android:layout_width="290dp"
    android:layout_height="55dp"
    android:layout_below="@id/line_2"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="32dp"
    android:background="@drawable/curved_rectangle"
    android:text="@string/sign_in"
    android:textColor="#FFFFFF"
    android:textSize="17sp" />

<TextView
    android:id="@+id/forgot_password"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/rectangle_1"
    android:layout_alignStart="@id/line_1"
    android:layout_marginStart="3dp"
    android:layout_marginTop="18dp"
    android:layout_marginBottom="30dp"
    android:text="@string/forgot_pass"
    android:textAppearance="@style/forgot_pass"
    android:textSize="15sp" />

<TextView
    android:id="@+id/create_account"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/rectangle_1"
    android:layout_alignEnd="@id/line_1"
    android:layout_marginTop="18dp"
    android:layout_marginEnd="3dp"
    android:layout_marginBottom="30dp"
    android:text="@string/create_acc"
    android:textAppearance="@style/create_acc"
    android:textSize="15sp" />

</RelativeLayout>
</ScrollView>

```

activity_notice.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"

```

```

android:layout_width="match_parent"
android:layout_height="wrap_content"
tools:context=".complaint.NoticeActivity">

```

```

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:paddingBottom="43dp">

```

```

<ImageView
android:id="@+id/arrow_1"
android:layout_width="35dp"
android:layout_height="35dp"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:layout_marginStart="30dp"
android:layout_marginTop="20dp"
android:src="@drawable/arrow_1"
tools:ignore="ContentDescription" />

```

```

<TextView
android:id="@+id/notice"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/arrow_1"
android:layout_alignParentStart="true"
android:layout_marginStart="30dp"
android:layout_marginTop="30dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="Notices"
android:textSize="28sp"
android:textColor="#000"/>

```

```

<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@id/notice"
android:layout_centerHorizontal="true"
android:layout_centerVertical="true"
android:layout_marginTop="27dp"
android:orientation="vertical"
android:padding="19dp">

```

```

<androidx.recyclerview.widget.RecyclerView
android:id="@+id/recycler_view"
android:layout_width="match_parent"
android:layout_height="wrap_content" />
</LinearLayout>
</RelativeLayout>
</androidx.coordinatorlayout.widget.CoordinatorLayout>

```

activity_notice_adapter_layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="wrap_content"

```

```

<androidx.cardview.widget.CardView
android:id="@+id/cardView1"
android:layout_width="wrap_content"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:layout_marginBottom="10dp"
app:cardCornerRadius="20dp"
app:cardElevation="4dp"
android:layout_marginLeft="30dp"
android:layout_marginRight="30dp"
android:layout_height="wrap_content">

```

```

<RelativeLayout
android:layout_width="wrap_content"
android:layout_height="wrap_content">

```

```

<ImageView
android:id="@+id/imageIcon"
android:layout_marginTop="5dp"
android:layout_marginLeft="10dp"
android:layout_marginRight="12dp"
android:src="@drawable/ganga_hostel_icon"
android:layout_width="40dp"
android:layout_height="40dp"
android:layout_alignParentTop="true"/>

```

```

<TextView
android:id="@+id/hostelMessName"
android:layout_marginTop="14dp"
android:textColor="#3F3F3F"
android:textSize="17sp"
android:fontFamily="@font/josefin_sans_semibold"
android:layout_marginRight="22dp"
android:layout_toRightOf="@id/imageIcon"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Ganga Hostel Mess" />

```

```

<TextView
android:id="@+id/date"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentRight="true"
android:fontFamily="@font/josefin_sans_semibold"
android:textFontWeight="500"
android:layout_marginTop="14dp"
android:textSize="15sp"
android:layout_marginRight="10dp"
android:textColor="#404040"
android:text="12-06-2021" />

```

```

<View
android:id="@+id/lineGanga"
android:layout_width="match_parent"
android:layout_height="2dp"
android:layout_marginTop="5dp"
android:layout_below="@id/imageIcon"
android:background="#D7D7D7"/>

```

```

<TextView
android:id="@+id/subject"
android:layout_width="wrap_content"

```

```

android:layout_height="wrap_content"
android:layout_below="@+id/lineGanga"
android:layout_marginTop="10dp"
android:textColor="#4D4D4D"
android:fontFamily="@font/josefin_sans_bold"
android:layout_marginLeft="15dp"
android:layout_marginRight="15dp"
android:text="Subject" />

```

```

<TextView
android:id="@+id/description"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/subject"
android:layout_marginTop="10dp"
android:textColor="#4D4D4D"
android:fontFamily="@font/josefin_sans_semibold"
android:textFontWeight="500"
android:layout_marginBottom="10dp"
android:layout_marginLeft="15dp"
android:layout_marginRight="15dp"
android:text="To all the students of Arya Mess :

```

Let me set the scene: your friend John is rushing to get his daughter from school and his son to the soccer field, and he still needs to stop at the grocery store because there's nothing in the fridge. In the midst of this everyday madness, he gets a text message from Google with a verification code. He thinks, "That's weird. Maybe I should log in to my email and see what's going on." />

```

<TextView
android:id="@+id/timeDisplay"
android:layout_alignParentRight="true"
android:layout_below="@id/description"
android:layout_marginRight="12dp"
android:layout_marginBottom="7dp"
android:textFontWeight="500"
android:layout_width="wrap_content"
android:textColor="#8A8A8A"
android:fontFamily="@font/josefin_sans_semibold"
android:layout_height="wrap_content"
android:text="02:30 PM" />

```

```

</RelativeLayout>
</androidx.cardview.widget.CardView>

```

```

</RelativeLayout>

```

activity_on_boarding.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".OnBoardingActivity"
android:padding="15dp">

<androidx.viewpager.widget.ViewPager
android:id="@+id/slider"
android:layout_width="match_parent"

```

```

android:layout_height="0dp"
app:layout_constraintBottom_toTopOf="@+id/relativeLayout"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
tools:layout_editor_absoluteX="1dp"
tools:layout_editor_absoluteY="1dp" >

```

```
</androidx.viewpager.widget.ViewPager>
```

```

<RelativeLayout
android:id="@+id/relativeLayout"
android:layout_width="match_parent"
android:layout_height="150dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent">

```

```

<TextView
android:id="@+id/Skip"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="28sp"
android:textColor="#898989"
android:text="Skip"
android:layout_alignParentLeft="true"
android:layout_alignParentBottom="true"
android:layout_marginLeft="40dp"
android:layout_marginBottom="45dp"

```

```
android:onClick="skip"/>
```

```

<LinearLayout
android:id="@+id/dots"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_above="@id/Skip"
android:layout_marginBottom="10dp"
android:layout_marginLeft="25dp"
android:orientation="horizontal">

```

```
</LinearLayout>
```

```

<Button
android:id="@+id/next_btn_slider"
android:layout_width="50dp"
android:layout_height="50dp"
android:layout_alignParentBottom="true"
android:layout_alignParentRight="true"
android:background="@drawable/slider_next_btn"
android:layout_marginBottom="40dp"
android:layout_marginRight="40dp"

```

```
android:onClick="next"/>
```

```

</RelativeLayout><?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"

```

```
android:layout_height="match_parent"
tools:context=".payment.PaymentDetailsActivity">
```

```
<RelativeLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:paddingBottom="40dp"
tools:ignore="MissingConstraints">
```

```
<ImageView
android:id="@id/arrow_1"
android:layout_width="35dp"
android:layout_height="35dp"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:layout_marginStart="30dp"
android:layout_marginTop="20dp"
android:src="@drawable/arrow_1"
tools:ignore="ContentDescription" />
```

```
<TextView
android:id="@+id/paymentDetails"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/arrow_1"
android:layout_alignParentStart="true"
android:layout_marginStart="30dp"
android:layout_marginTop="30dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="@string/last_payment_details"
android:textSize="28sp"
android:textColor="#000"/>
```

```
<LinearLayout
android:layout_width="316dp"
android:layout_height="match_parent"
android:layout_below="@id/paymentDetails"
android:layout_centerHorizontal="true"
android:layout_centerVertical="true"
android:layout_marginTop="27dp"
android:background="#FFFFFF"
android:orientation="vertical"
android:padding="5dp">
```

```
<androidx.recyclerview.widget.RecyclerView
android:id="@+id/recycler_view"
android:layout_width="match_parent"
android:layout_height="wrap_content"
tools:listitem="@layout/item_payment_details"/>
</LinearLayout>
```

```
</RelativeLayout>
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_payment_menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
```

```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".payment.PaymentMenuActivity">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/mess_pay_chart"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="9" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingTop="5dp"
        android:paddingBottom="5dp"
        android:paddingLeft="15sp"
        android:gravity="start"
        android:textStyle="bold"
        android:text="Total: 30Rs"
        android:textSize="20sp"
        android:textColor="@color/yellow"
        android:id="@+id/total_amt"
        android:visibility="gone"
    />

    <Button
        android:id="@+id/pay_meal"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:textSize="20sp"
        android:backgroundTint="@color/main_colour"
        android:textColor="@color/white"
        android:text="Pay" />
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_payment_details.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".payment.PaymentDetailsActivity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```
tools:ignore="MissingConstraints">
```

```
<ImageView
    android:id="@id/arrow_1"
    android:layout_width="35dp"
    android:layout_height="35dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="30dp"
    android:layout_marginTop="20dp"
    android:src="@drawable/arrow_1"
    tools:ignore="ContentDescription" />
```

```
<TextView
    android:id="@+id/paymentDetails"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/arrow_1"
    android:layout_alignParentStart="true"
    android:layout_marginStart="30dp"
    android:layout_marginTop="30dp"
    android:fontFamily="@font/josefin_sans_bold"
    android:text="@string/last_payment_details"
    android:textSize="28sp"
    android:textColor="#000"/>
```

```
<LinearLayout
    android:layout_width="316dp"
    android:layout_height="match_parent"
    android:layout_below="@id/paymentDetails"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:layout_marginTop="27dp"
    android:background="#FFFFFF"
    android:orientation="vertical"
    android:padding="5dp">
```

```
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recycler_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    tools:listitem="@layout/item_payment_details"/>
</LinearLayout>
```

```
</RelativeLayout>
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_q_r_scanner.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".home.QRScanner">
```

```
<com.budiyev.android.codescanner.CodeScannerView
    android:id="@+id/scanner"
    android:layout_width="wrap_content"
```



```

app:autoFocusButtonColor="@android:color/black"
app:autoFocusButtonVisible="true"
app:flashButtonColor="@android:color/white"
app:flashButtonVisible="true"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.5"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />

```

```

<TextView
    android:id="@+id/result"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="12dp"
    android:gravity="center"
    android:text="Text View"
    android:textSize="24sp"
    android:textColor="#fff"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.666"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.953" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_qrscanner.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".home.QRScanner">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="QRScanner"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_splash_screen.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```

android:gravity="center"
android:orientation="vertical"
android:background="@color/profileBackground"
tools:context=".Splash_screen">

```

```

<ImageView
android:id="@+id/logo"
android:layout_width="200dp"
android:layout_height="200dp"
android:layout_gravity="center"

android:layout_marginStart="40dp"
android:layout_marginEnd="50dp"
android:src="@drawable/logo"
tools:ignore="ContentDescription" />

```

```

</LinearLayout>

```

activity_vote.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".vote.VoteActivity">

```

```

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:paddingBottom="43dp">

```

```

<ImageView
android:id="@id/arrow_3"
android:layout_width="35dp"
android:layout_height="35dp"
android:layout_marginStart="25dp"
android:layout_marginTop="20dp"
android:src="@drawable/arrow_1"
tools:ignore="ContentDescription" />

```

```

<TextView
android:id="@+id/vote_for_ch"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/arrow_3"
android:layout_marginStart="25dp"
android:textSize="18sp"
android:layout_marginTop="30dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="Vote for Change"/>

```

```

<TextView
android:id="@+id/choose_meal"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/vote_for_ch"
android:layout_marginStart="25dp"
android:layout_marginTop="60dp"
android:fontFamily="@font/josefin_sans_semibold"
android:text="Choose Meal" />

```

```

<RadioGroup
    android:id="@+id/radio_bottom_group"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/choose_meal"
    android:layout_marginStart="36dp"
    android:layout_marginTop="19dp">

```

```

<RadioButton
    android:id="@+id/breakfast"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:button="@drawable/radio_button_selector"
    android:fontFamily="@font/josefin_sans_semibold"
    android:padding="8dp"
    android:text="Breakfast"
    android:checked="true"/>

```

```

<RadioButton
    android:id="@+id/lunch"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="9dp"
    android:button="@drawable/radio_button_selector"
    android:fontFamily="@font/josefin_sans_semibold"
    android:padding="8dp"
    android:text="Lunch" />

```

```

<RadioButton
    android:id="@+id/dinner"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="9dp"
    android:button="@drawable/radio_button_selector"
    android:fontFamily="@font/josefin_sans_semibold"
    android:padding="8dp"
    android:text="Dinner" />
</RadioGroup>

```

```

<TextView
    android:id="@+id/food_item"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/radio_bottom_group"
    android:layout_marginStart="25dp"
    android:layout_marginTop="45dp"
    android:fontFamily="@font/josefin_sans_semibold"
    android:text="Food item" />

```

```

<EditText
    android:id="@+id/enter_food_item"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/food_item"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="30dp"
    android:backgroundTint="#39374D"
    tools:ignore="Autofill,LabelFor,TextFields" />

```

```

<Button
    android:id="@+id/vote_for_ad"
    android:layout_width="130dp"
    android:layout_height="37dp"

```

```

android:layout_marginStart="25dp"
android:layout_marginTop="65dp"
android:background="@drawable/white_background"
android:elevation="8dp"
android:textSize="10sp"
android:fontFamily="@font/josefin_sans_semibold"
android:text="Vote for adding" />

```

```

<Button
android:id="@+id/vote_for_re"
android:layout_width="130dp"
android:layout_height="37dp"
android:layout_below="@id/enter_food_item"
android:layout_alignParentEnd="true"
android:layout_marginTop="65dp"
android:layout_marginEnd="30dp"
android:background="@drawable/white_background"
android:elevation="8dp"
android:textSize="10sp"
android:fontFamily="@font/josefin_sans_semibold"
android:text="Vote for Removing" />

```

```

<Button
android:id="@+id/submit_button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/vote_for_ad"
android:layout_marginStart="35dp"
android:layout_marginTop="85dp"
android:layout_marginEnd="35dp"
android:background="@drawable/rectangle_1"
android:elevation="8dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="Submit"
android:textColor="#fff"/>
</RelativeLayout>
</ScrollView>

```

fragment_complaint.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".home.ComplaintFragment">

```

```

<!-- <Button-->
<!-- android:id="@+id/compBtn"-->
<!-- android:layout_width="wrap_content"-->
<!-- android:layout_height="wrap_content"-->
<!-- android:background="#149548"-->
<!-- android:text="File a Complaint"-->
<!-- android:textAllCaps="true"-->
<!-- android:textSize="24sp"-->
<!-- android:textStyle="bold"-->
<!-- app:layout_constraintBottom_toBottomOf="parent"-->
<!-- app:layout_constraintEnd_toEndOf="parent"-->
<!-- app:layout_constraintHorizontal_bias="0.5"-->
<!-- app:layout_constraintStart_toStartOf="parent"—

```

```
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:paddingBottom="43dp">
```

```
<EditText
    android:id="@+id/editTextTextPersonName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/rectangle_4"
    android:layout_alignTop="@+id/rectangle_4"
    android:layout_alignEnd="@+id/rectangle_4"
    android:layout_marginStart="0dp"
    android:layout_marginTop="0dp"
    android:layout_marginEnd="0dp"
    android:background="@null"
    android:ems="10"
    android:scrollbars="vertical"
    android:inputType="textMultiLine"
    android:maxLines="7" />
```

```
<TextView
    android:id="@+id/complaint_s"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginTop="30dp"
    android:textSize="28sp"
    android:textColor="#000"
    android:fontFamily="@font/josefin_sans_bold"
    android:text="Complaint Section" />
```

```
<TextView
    android:id="@+id/issue"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/complaint_s"
    android:layout_marginStart="25dp"
    android:layout_marginTop="52dp"
    android:textColor="#000"
    android:fontFamily="@font/josefin_sans_bold"
    android:text="Issue"/>
```

```
<EditText
    android:id="@+id/line_1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/issue"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="30dp"
    android:backgroundTint="#39374D"
    android:inputType="text"
    android:textColor="#000000"
    tools:ignore="Autofill,LabelFor" />
```

```
<TextView
    android:id="@+id/description"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/line_1"
    android:layout_marginStart="25dp"
```

```

android:layout_marginTop="28dp"
android:fontFamily="@font/josefin_sans_bold"
android:text="Description"
android:textColor="#000"/>
<EditText
android:id="@+id/line_2"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/description"
android:layout_marginStart="25dp"
android:layout_marginEnd="30dp"
android:backgroundTint="#39374D"
android:inputType="text"
android:textColor="#000000"
tools:ignore="Autofill,LabelFor" />

```

```

<TextView
android:id="@+id/img"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/line_2"
android:layout_marginStart="25dp"
android:layout_marginTop="28dp"
android:textColor="#000"
android:fontFamily="@font/josefin_sans_bold"
android:text="Complaint Image" />

```

```

<ImageView
android:id="@+id/rectangle_4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/img"
android:layout_centerHorizontal="true"
android:layout_marginStart="25dp"
android:layout_marginEnd="30dp"
android:scaleType="fitXY"
android:src="@drawable/description_box"
tools:ignore="ContentDescription" />

```

```

<Button
android:id="@+id/upload_button"
android:layout_width="135dp"
android:layout_height="38dp"
android:layout_below="@id/rectangle_4"
android:layout_marginStart="25dp"
android:layout_marginTop="24dp"
android:background="@drawable/white_background"
android:elevation="12dp"
android:fontFamily="@font/josefin_sans_bold"
android:gravity="center"
android:text="Upload Image"
android:textColor="#000"/>

```

```

<Button
android:id="@+id/file_complaint_button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/upload_button"
android:layout_marginStart="35dp"
android:layout_marginTop="80dp"
android:layout_marginEnd="35dp"
android:background="@drawable/rectangle_"

```

```

android:fontFamily="@font/josefin_sans_bold"
android:text="File Complaint"
android:textColor="#fff" />
</RelativeLayout>
</ScrollView>

```

fragment_home.xml

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    tools:context=".home.HomeFragment">
    <!-- TODO: Update blank fragment layout -->
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <TextView
            android:id="@+id/name_tv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Good morning Priyanshu!"
            android:textColor="#000000"
            android:fontFamily="sans-serif-medium"
            android:layout_marginTop="16dp"
            android:layout_marginLeft="16dp"
            android:textStyle="bold"
            android:textSize="20sp" />
        <TextView
            android:id="@+id/meals_tv"
            android:layout_below="@id/name_tv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Upcoming Meals"
            android:textColor="#000000"
            android:fontFamily="sans-serif"
            android:layout_marginTop="8dp"
            android:layout_marginLeft="16dp"
            android:textSize="12sp" />

        <TextView
            android:id="@+id/txt"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Bon Appetit"
            android:layout_below="@id/meal_next_button"
            android:textColor="#000000"

            android:layout_marginLeft="16dp" />

        <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/meals_list"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:scrollbars="horizontal"
            android:layout_marginVertical="5dp"
            android:layout_below="@id/meals_tv" >
    </androidx.recyclerview.widget.RecyclerView>

```

```

<TextView
    android:id="@+id/meal_next_button"
    android:layout_width="80dp"
    android:layout_height="wrap_content"
    android:layout_below="@id/meals_list"
    android:layout_alignParentEnd="true"
    android:layout_marginTop="12dp"
    android:layout_marginEnd="10dp"
    android:background="@drawable/curved_rectangle"
    android:fontFamily="@font/josefin_sans_bold"
    android:padding="10dp"
    android:textAlignment="center"
    android:text="Next"
    android:textColor="#fff" />
<Spinner
    android:id="@+id/spinner"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/meals_list"
    android:layout_alignParentStart="true"
    android:layout_marginTop="12dp"
    android:layout_marginStart="10dp"
/>

```

```

<androidx.cardview.widget.CardView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/txt"

```

```

    android:layout_margin="10dp"
    app:cardCornerRadius="20dp"
    app:cardElevation="10dp">

```

```

<RelativeLayout
    android:id="@+id/todays_special"
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:layout_centerHorizontal="true">
    <!--        android:elevation="6dp">-->
    <!--        android:layout_marginHorizontal="10dp"-->
    <!--        android:layout_above="@+id/text"-->
    <!--        android:layout_below="@id/txt"-->
    <!--        android:layout_marginTop="16dp"-->
    <!--        android:layout_marginBottom="79dp"-->
    <ImageView
        android:id="@+id/food_img"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:scaleType="fitXY"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="10dp"
        android:src="@drawable/food_img" />

```

```

<TextView
    android:id="@+id/delight_tv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/food_img"
    android:layout_marginStart="22dp"
    android:text="Today's Delight."
    android:textColor="@android:color/darker_gray"
    android:textSize="12sp" />

```



```

<TextView
    android:id="@+id/food_name_tv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/delight_tv"
    android:layout_marginStart="22dp"
    android:layout_marginTop="4dp"
    android:text="Panner Butter Masala"
    android:textColor="#000000"
    android:textSize="16sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/food_description"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/food_name_tv"
    android:layout_marginStart="22dp"
    android:layout_marginTop="4dp"
    android:text="Detail of food Item"
    android:textColor="#000000"
    android:textSize="13sp" />

</RelativeLayout>
</androidx.cardview.widget.CardView>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/text"
    android:layout_alignParentBottom="true"/>

</RelativeLayout>
</FrameLayout>

```

fragment_profile.xml

```

<?xml version="1.0" encoding="utf-8"?>

<!-- Here you put the rest of your current view-->
<!--suppress AndroidUnresolvableTag -->
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_centerInParent="false"
    android:background="@color/profileBackground"
    android:gravity="center_horizontal"
    android:orientation="vertical"
    tools:context=".home.ProfileFragment">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:padding="10dp">

        <TextView
            android:layout_width="0dp"

```

```

android:layout_height="fill_parent"
android:layout_gravity="left"
android:layout_marginStart="10dp"
android:layout_marginLeft="10dp"
android:layout_marginTop="10dp"
android:layout_weight="8"
android:text="@string/profile1"
android:textSize="24sp"
android:textStyle="bold" />
</LinearLayout>

```

```

<ImageView
android:id="@+id/profile_img"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:scaleType="centerInside"
android:src="@drawable/undraw_profile_pic" />

```

```

<TextView
android:id="@+id/name"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
tools:text="@string/xyz"
android:textSize="24sp"
android:textStyle="bold" />

```

```

<LinearLayout
android:layout_width="match_parent"
android:layout_height="205dp"
android:background="@drawable/undraw_cap"
android:orientation="vertical">

```

```

<TextView
android:id="@+id/email"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center_horizontal"
tools:text="abc@xyz.com"
android:textSize="18sp"
android:textStyle="bold" />

```

```

<TextView
android:id="@+id/Room_no"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center_horizontal"
android:layout_marginTop="10dp"
tools:text="@string/room_no_339"
android:textSize="24sp"
android:textStyle="bold" />
</LinearLayout>

```

```

<Button
android:id="@+id/edit_profile_btn"
style="@style/profileStyledBackButton"
android:layout_marginTop="20dp"
android:gravity="center"
android:padding="5dp"
android:text="@string/edit_profile"
android:textColor="@color/profileBackground" />

```

```

<Button
    android:id="@+id/view_coupon_btn"
    style="@style/profileStyledBackButton"
    android:layout_marginTop="10dp"
    android:gravity="center"
    android:padding="5dp"<?xml version="1.0" encoding="utf-8"?>
    <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/scanButton"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="24dp"
        tools:context=".home.QRFragment">

```

```

<ImageView
    android:id="@+id/imageViewQR"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_marginBottom="32dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.328" />

```

```

<TextView
    android:id="@+id/nameID"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="49dp"
    android:layout_marginBottom="240dp"
    android:text="Sameer"
    android:textColor="#000"
    android:textSize="24sp"
    android:textStyle="bold|italic"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent" />

```

```

<TextView
    android:id="@+id/text View"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="94dp"
    android:text="e-Coupon"
    android:textStyle="italic"
    android:textSize="20sp"
    android:textColor="#000"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
    android:text="@string/view_coupon"
    android:textColor="@color/profileBackground" />
</LinearLayout>

```

fragment_q_r.xml

```

<?xml version="1.0" encoding="utf-8"?>

```

```

xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/scanButton"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:padding="24dp"
tools:context=".home.QRFragment">

```

```

<ImageView
android:id="@+id/imageViewQR"
android:layout_width="200dp"
android:layout_height="200dp"
android:layout_marginBottom="32dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.498"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.328" />

```

```

<TextView
android:id="@+id/nameID"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="49dp"
android:layout_marginBottom="240dp"
android:text="Sameer"
android:textColor="#000"
android:textSize="24sp"
android:textStyle="bold|italic"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintStart_toStartOf="parent" />

```

```

<TextView
android:id="@+id/textView"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="32dp"
android:layout_marginTop="94dp"
android:text="e-Coupon"
android:textStyle="italic"
android:textSize="20sp"
android:textColor="#000"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

item_payment_details.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_margin="8dp"
android:background="@drawable/white_background"
xmlns:tools="http://schemas.android.com/tools"
android:orientation="vertical">

```

```

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">

```

```

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Date: "
android:fontFamily="@font/josefin_sans_bold"
android:textSize="26sp"
android:textColor="#000"/>
<TextView
android:id="@+id/date"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="22sp"
android:fontFamily="@font/josefin_sans_bold"
tools:text="Wed May 12 00:00:00 GMT+5:30 2021"/>

```

```

</LinearLayout>

```

```

<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">

```

```

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Session: "
android:fontFamily="@font/josefin_sans_bold"
android:textSize="26sp"
android:textColor="#000"/>
<TextView
android:id="@+id/session"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="22sp"
android:fontFamily="@font/josefin_sans_bold"
tools:text="LUNCH"/>

```

```

</LinearLayout>

```

```

<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">

```

```

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Status: "
android:fontFamily="@font/josefin_sans_bold"
android:textSize="26sp"
android:textColor="#000"/>
<TextView
android:id="@+id/status"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
tools:text="Paid"
android:textAllCaps="true"
android:textSize="22sp"

```

```
</LinearLayout>
```

```
</LinearLayout>
```

item_view.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="@drawable/white_background"
    xmlns:tools="http://schemas.android.com/tools">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        tools:text="sample notice"
        android:id="@+id/text_view"/>

</LinearLayout>
```

meals_list_item.xml

```
<?xml version="1.0" encoding="utf-8"?>

<androidx.cardview.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/meal_card"
    android:layout_gravity="fill"
    android:layout_margin="8dp"
    app:cardBackgroundColor="@color/profileBackground"
    app:cardCornerRadius="10dp"
    app:cardElevation="40dp">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:clickable="true"
        android:padding="10dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/imageView_meal"
            android:layout_w
            android:layout_height="60dp"
            android:layout_centerHorizontal="true"
            android:scaleType="centerCrop"
            android:src="@drawable/breakfast" />

        <TextView
            android:id="@+id/textView_meal"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/imageView_meal"
            android:layout_centerHorizontal="true"
            android:text="Meal Name"
```

```

android:textAlignment="center"
android:textColor="@color/colorPrimaryDark"
android:textStyle="bold" />
</RelativeLayout>
</androidx.cardview.widget.CardView>

```

nav_header.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#373549"
    >
    <ImageView
        android:layout_marginTop="30dp"
        android:padding="18dp"
        android:layout_width="300dp"
        android:layout_gravity="center"
        android:layout_height="150dp"
        android:src="@drawable/logo"
    />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="messsy"
        android:textStyle="bold|italic"
        android:textColor="#fff"
        android:layout_gravity="center"
        android:textSize="40sp"/>
    <View
        android:layout_width="220dp"
        android:layout_height="1dp"
        android:layout_gravity="center"
        android:background="#fff"/>

</LinearLayout>

```

payment_menu_element.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <de.hdodenhof.circleimageview.CircleImageView
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:id="@+id/element_img"
        android:src="@drawable/scan"
        android:layout_margin="10dp"
    />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/element_title"
        android:text="Title"
        android:layout_toEndOf="@id/element_img"
    />
</RelativeLayout>

```

```

android:layout_marginTop="10dp"
android:textColor="@color/black"
/>
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:id="@+id/element_price"
android:text="Price: 00"
android:layout_toEndOf="@id/element_img"
android:textSize="20sp"
android:layout_below="@id/element_title"
android:textColor="@color/main_colour"
/>
<Button
android:layout_width="40dp"
android:layout_height="40dp"
android:text="-"
android:id="@+id/element_minus"
android:layout_toStartOf="@id/element_quantity"
android:layout_centerVertical="true"
/>
<TextView
android:layout_width="40dp"
android:layout_height="40dp"
android:text="0"
android:textColor="@color/black"
android:id="@+id/element_quantity"
android:layout_toStartOf="@id/element_plus"
android:layout_centerVertical="true"
android:gravity="center"
/>
<Button
android:layout_width="40dp"
android:layout_height="40dp"
android:text="+"
android:layout_alignParentEnd="true"
android:layout_centerVertical="true"
android:layout_marginEnd="10dp"
android:id="@+id/element_plus" />
</RelativeLayout>

```

slides_layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent">

<ImageView
android:id="@+id/slider_image"
android:layout_width="0dp"
android:layout_height="350dp"
android:padding="35dp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:srcCompat="@drawable/slider_foeding_made_easier" />

<TextView

```



```

android:id="@+id/slider_heading"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_margin="5dp"
android:paddingLeft="25dp"
android:text="@string/first_slide_title"
android:textColor="#000000"
android:textSize="30sp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/slider_image" />

```

```

<TextView
android:id="@+id/slider_description"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginLeft="9dp"
android:paddingLeft="25dp"
android:layout_marginRight="5dp"
android:text="@string/first_slide_desc"
android:textColor="#767676"
app:layout_constraintTop_toBottomOf="@id/slider_heading"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
android:textSize="24sp"
android:paddingTop="15dp" />

```

```

</androidx.constraintlayout.widget.ConstraintLayout>

```

upcoming_meal.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical" android:layout_width="match_parent"
android:layout_height="match_parent">

</LinearLayout>

```

8.2 Java coding

Complaint.java

```

package in.hackslash.messsy.complaint;

public class Complaint {
    private String issue;
    private String description;
    private String image;
    private String id;
    public Complaint(String issue,String description,String id){
        this.issue=issue;
        this.description=description;
        this.id=id;
    }

    public String getIssue() {
        return issue;
    }
}

```

```

public String getId() {
    return id;
}

public String getDescription() {
    return description;
}
// TODO Add complaint fields here
}

```

ComplaintActivity.java

```

package in.hackslash.messsy.complaint;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import android.widget.Toolbar;

import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import in.hackslash.messsy.R;

public class ComplaintActivity extends AppCompatActivity {

    // TODO Declare instances for all UI elements
    EditText issue;
    EditText des;
    ImageView imgView;
    private static final int IMAGE_CODE=1;
    Button upload,submit;
    Uri downloadURL;
    StorageReference mstorageReference;
    FirebaseFirestore dbs = FirebaseFirestore.getInstance();
    private final CollectionReference complaintRef = dbs.collection("ComplaintSection");

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_complaint);
        issue=findViewById(R.id.line_1); des=findViewById

```

```
upload=findViewById(R.id.upload_button);
```

```
submit=findViewById(R.id.file_complaint_button1);
// submit=findViewById(R.id.submit_button);
```

```
mstorageReference= FirebaseStorage.getInstance().getReference().child("images");
```

```
setup();
}
```

```
public void setup() {
// TODO define onclicklistener on File Complaints button to collect data from all edittexts and call
ComplaintsUtil.createComplaint
submit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

f(v);

}
});
// TODO Show an indefinite progress bar on the screen as soon as the submit button is clicked to denote that data is
being processed
upload.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent=new Intent().setAction(Intent.ACTION_GET_CONTENT);
intent.setType("image/*");
startActivityForResult(Intent.createChooser(intent,"Complete action using"), IMAGE_CODE);

}
});
// TODO Pass a callback by createComplaint().setOnCompleteListener() to handle the result from the create
operation. Show a toast according to the result and also hide the progress bar.
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
super.onActivityResult(requestCode, resultCode, data);
if(requestCode==IMAGE_CODE&& resultCode==RESULT_OK){
assert data != null;
Uri selected=data.getData();
try {
Bitmap currentImage = MediaStore.Images.Media.getBitmap
(this.getContentResolver(), selected);

imgView.setImageBitmap(currentImage);
}
catch (Exception e){
e.printStackTrace();
}
StorageReference photoRef = mstorageReference.child(selected.getLastPathSegment());
Toast.makeText(ComplaintActivity.this,"Loading",Toast.LENGTH_SHORT).show();
```

```

@Override
public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
    Task<Uri> urlTask = taskSnapshot.getStorage().getDownloadUrl();
    while (!urlTask.isSuccessful());
    downloadURL = urlTask.getResult();

}

});
}
}
public void f(View v){
    String s = String.valueOf(downloadURL);
    String s1=issue.getText().toString();
    String s2=des.getText().toString();
    if(s==null||s1==null||s2==null) {
        Complaint complaint = new Complaint(s1, s2, s);

        complaintRef.add(complaint);

        Toast.makeText(this, "complaint added", Toast.LENGTH_SHORT).show();
    }
    else
        Toast.makeText(this, "enter all the fields", Toast.LENGTH_SHORT).show();

}
}

```

ComplaintsUtil.java

```

package in.hackslash.messsy.complaint;

public class ComplaintsUtil {

    private onCompleteListener onCompleteListener;

    public ComplaintsUtil createComplaint(Complaint complaint){

        // TODO Check if a user is logged in and return NO_USER if no logged in user is found

        // TODO Validate complaint here and return WRONG_INPUT if anything is wrong

        // TODO Check DB if the complaint exists and return DUPLICATE_COMPLAINT if complaint exists

        // TODO Create new complaint using
        FirebaseDatabase.getInstance().getReference().child('complaints/').push().set(complaint) and return SUCCESS if
        successful

        // Leave as it is
        return this;
    }

    public ComplaintsUtil fetchNotice() {

        // TODO Check DB if a notice exists and return NOTICE_DOES_NOT_EXIST if notice is not found

        // TODO Return the notice in a Notice object if found with a SUCCESS Status
        // Leave as it is
    }
}

```

```
return this;
```

```
}
```

```
public void setOnCompleteListener(ComplaintsUtil.onCompleteListener onCompleteListener) {
    this.onCompleteListener = onCompleteListener;
}
```

```
public interface onCompleteListener {
    void onComplete(Status status, Complaint complaint);
}
```

```
public enum Status {
    SUCCESS, DUPLICATE_COMPLAINT, WRONG_INPUT, NOTICE_DOES_NOT_EXIST, NO_USER
}
}
```

Notice.java

```
package in.hackslash.messsy.complaint;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
```

```
import java.util.ArrayList;
```

```
import in.hackslash.messsy.R;
```

```
public class Notice extends RecyclerView.Adapter<viewholder> {
```

```
    ArrayList<NoticeBoxAdapterClass> arrayList = new ArrayList<>();
    Notice( NoticeActivity noticeActivity){
```

```
}
```

```
@NonNull
```

```
@Override
```

```
public viewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_notice_adapter_layout, parent, false);
    viewholder v1 = new viewholder(view);
```

```
return v1;
```

```
}
```

```
void updatefood(ArrayList<NoticeBoxAdapterClass> f){
    // arrayList.clear();
    arrayList.addAll(f);
    notifyDataSetChanged();
}
```

```
@Override
```

```
public void onBindViewHolder(@NonNull viewholder holder, int position) {
    holder.subject.setText(arrayList.get(position).getSubject());
    holder.description.setText(arrayList.get(position).getDescription());
    holder.hostelMessName.setText(arrayList.get(position).getHostel() + " " + arrayList.get(position).getDesignation());
    String hostel = arrayList.get(position).getHostel().toString();if("brahmputra".equalsIgnoreCase(hostel))
```

```

{
holder.imageView.setImageResource(R.drawable.brahmaputra_hostel_icon);
}
else if("kosi".equalsIgnoreCase(hostel))
{
holder.imageView.setImageResource(R.drawable.kosi_icon);
}
else if("ganga".equalsIgnoreCase(hostel))
{
holder.imageView.setImageResource(R.drawable.ganga_hostel_icon);
}
else if("sone".equalsIgnoreCase(hostel))
{
holder.imageView.setImageResource(R.drawable.sone_icon);
}
else
{
holder.imageView.setImageResource(R.drawable.admin_profile_fragment_img);
}
holder.date.setText(arrayList.get(position).getDate());
holder.time.setText(arrayList.get(position).getTimestamp());
}

@Override
public int getItemCount() {
return arrayList.size();
}

}

// TODO Add notice fields here
}
class viewholder extends RecyclerView.ViewHolder{
ImageView imageView;
TextView hostelMessName, description, date, time, subject;
public viewholder(@NonNull View itemView) {
super(itemView);
imageView = itemView.findViewById(R.id.imageIcon);
hostelMessName = itemView.findViewById(R.id.hostelMessName);
description = itemView.findViewById(R.id.description);
date = itemView.findViewById(R.id.date);
time = itemView.findViewById(R.id.timeDisplay);
subject = itemView.findViewById(R.id.subject);
}
}
}

```

NoticeActivity.java

```

package in.hackslash.messsy.complaint;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.ImageView;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

```

```

import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

import java.util.ArrayList;

import in.hackslash.messsy.R;
import in.hackslash.messsy.home.HomeActivity;

public class NoticeActivity extends AppCompatActivity {

    // TODO Declare instances for all UI elements

    RecyclerView r;
    ImageView imageView;

    String date, description, subject, time, user, hostel, designation;

    private final String TAG = "abcd";
    private final FirebaseFirestore db = FirebaseFirestore.getInstance();
    private final CollectionReference notebookRef = db.collection("MessssyNotice");
    Notice madapter = new Notice(this);
    ArrayList<NoticeBoxAdapterClass> notice = new ArrayList<NoticeBoxAdapterClass>();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_notice);
        r = findViewById(R.id.recycler_view);
        imageView=findViewById(R.id.arrow_1);
        imageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                startActivity(new Intent(NoticeActivity.this, HomeActivity.class));
            }
        });
        RecyclerView.LayoutManager layoutManager;
        layoutManager = new LinearLayoutManager(this, LinearLayoutManager.VERTICAL, false);
        r.setLayoutManager(layoutManager);
        load();
        r.setAdapter(madapter);
    }

    public void load() {
        notebookRef.get().addOnSuccessListener(new OnSuccessListener<QuerySnapshot>() {
            @Override
            public void onSuccess(QuerySnapshot queryDocumentSnapshots) {

                for (QueryDocumentSnapshot documentSnapshot : queryDocumentSnapshots) {
                    date = documentSnapshot.getString("date");
                    description = documentSnapshot.getString("description");
                    subject = documentSnapshot.getString("subject");
                    time = documentSnapshot.getString("timestamp");
                    user = documentSnapshot.getString("userInfo");
                    hostel = documentSnapshot.getString("hostel");
                    designation = documentSnapshot.getString("designation");

                    notice.add(new NoticeBoxAdapterClass(subject, description, date, time, designation, hostel));
                }
            }
        });
    }
}

```

```

madapter.updatefood(notice);
}
}).addOnFailureListener(new OnFailureListener() {
@Override
public void onFailure(@NonNull Exception e) {
Log.d(TAG, e.toString());

}
});
}
}

```

NoticeBoxAdapterClass.java

```

package in.hackslash.messsy.complaint;

public class NoticeBoxAdapterClass
{

private String subject;
private String Description;

String timestamp;
private String date, designation, hostel;

public NoticeBoxAdapterClass (String subject, String Description, String date, String timestamp, String designation,
String hostel){

this.subject=subject;
this.Description= Description;
this.date=date;
this.timestamp=timestamp;
this.designation=designation;
this.hostel=hostel;

}
public NoticeBoxAdapterClass ()
{

}

public String getSubject() {
return subject;
}

public void setSubject(String subject) {
this.subject = subject;
}

public String getDescription() {
return Description;
}

public void setDescription(String description) {
Description = description;
}

```



```

}

public String getTimestamp() {
return timestamp;
}

public void setTimestamp(String timestamp) {
this.timestamp = timestamp;
}

public String getDate() {
return date;
}

public void setDate(String date) {
this.date = date;
}

public String getDesignation() {
return designation;
}

public void setDesignation(String designation) {
this.designation = designation;
}

public String getHostel() {
return hostel;
}

public void setHostel(String hostel) {
this.hostel = hostel;
}

}

```

MealData.java

```

package in.hackslash.messsy.home.home;

public class MealData {
String imageUrl;
String name;
String description;
int cardColor;
int textColor;

public MealData(String imageUrl, String description, String name,int cardColor,int textColor) {
this.imageUrl = imageUrl;
this.description = description;
this.name = name;
this.cardColor = cardColor;
this.textColor=textColor;
}

public MealData() {
}
public int getCardColor(){return cardColor;}
public void setCardColor(int cardColor){
this.cardColor = cardColor;
}
public int getTextColor(){return textColor;}
}

```

```

public String getImageUrl() {
return imageUrl;
}

public void setImageUrl(String imageUrl) {
this.imageUrl = imageUrl;
}

public String getDescription() {
return description;
}

public void setDescription(String description) {
this.description = description;
}

public String getName() {
return name;
}

public void setName(String name) {
this.name = name;
}
}

```

MealListAdapter.java

```

package in.hackslash.messsy.home.home;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.cardview.widget.CardView;
import androidx.recyclerview.widget.RecyclerView;

import com.squareup.picasso.Picasso;

import java.util.ArrayList;

import in.hackslash.messsy.R;

public class MealListAdapter extends RecyclerView.Adapter<MealListAdapter.ViewHolder> {

    ArrayList<MealData> mealList;
    Context context;
    public MealListAdapter(Context context, ArrayList<MealData> mealList) {
        this.context = context;
        this.mealList = mealList;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(context).inflate(R.layout.meals_list_item, parent, false);
        final ViewHolder holder=new ViewHolder(view);
        return holder;
    }

```

```

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
    if (mealList != null && mealList.size() > 0) {
        MealData model = mealList.get(position);
        holder.mealname.setText(model.getName());
        Picasso.with(context).load(model.getImageUrl()).into(holder.image);
        holder.card.setCardBackgroundColor(context.getResources().getColor(model.getCardColor()));
        holder.mealname.setTextColor(context.getResources().getColor(model.getTextColor()));
    } else {
        return;
    }
}

```

```

@Override
public int getItemCount() {
    return mealList.size();
}

```

```

public class ViewHolder extends RecyclerView.ViewHolder {
    TextView mealname, description;
    ImageView image;
    CardView card;

```

```

    public ViewHolder(@NonNull View itemView) {
        super(itemView);
        card=itemView.findViewById(R.id.meal_card);
        image = itemView.findViewById(R.id.imageView_meal);
        mealname = itemView.findViewById(R.id.textView_meal);
        description = itemView.findViewById(R.id.textView_meal);
    }
}

```

RecyclerViewItemClickListener.java

```

package in.hackslash.messsy.home.home;

```

```

import android.content.Context;
import android.view.GestureDetector;
import android.view.MotionEvent;
import android.view.View;

```

```

import androidx.recyclerview.widget.RecyclerView;

```

```

public class RecyclerViewItemClickListener implements RecyclerView.OnItemTouchListener {
    private OnItemClickListener mListener;

```

```

    public interface OnItemClickListener {
        public void onItemClick(View view, int position);

```

```

    public void onLongItemClick(View view, int position);
    }

```

```

    GestureDetector mGestureDetector;

```

```

    public RecyclerViewItemClickListener(Context context, final RecyclerView recyclerView, OnItemClickListener listener) {
        mListener = listener;
        mGestureDetector = new GestureDetector(context, new GestureDetector.SimpleOnGestureListener() {

```

```

@Override
public boolean onSingleTapUp(MotionEvent e) {
    return true;
}

@Override
public void onLongPress(MotionEvent e) {
    View child = recyclerView.findChildViewUnder(e.getX(), e.getY());
    if (child != null && mListener != null) {
        mListener.onLongItemClick(child, recyclerView.getChildAdapterPosition(child));
    }
}

@Override public boolean onInterceptTouchEvent(RecyclerView view, MotionEvent e) {
    View childView = view.findChildViewUnder(e.getX(), e.getY());
    if (childView != null && mListener != null && mGestureDetector.onTouchEvent(e)) {
        mListener.onItemClick(childView, view.getChildAdapterPosition(childView));
        return true;
    }
    return false;
}

@Override public void onTouchEvent(RecyclerView view, MotionEvent motionEvent) { }

@Override
public void onRequestDisallowInterceptTouchEvent (boolean disallowIntercept){ }
}

```

ComplaintFragment.java

```

package in.hackslash.messsy.home;

import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.content.Intent;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;

import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

```

```

import in.hackslash.messsy.R;
import in.hackslash.messsy.complaint.Complaint;

import androidx.fragment.app.Fragment;

import in.hackslash.messsy.R;
import in.hackslash.messsy.complaint.ComplaintActivity;

/**
 * A simple { @link Fragment} subclass.
 * Use the { @link ComplaintFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class ComplaintFragment extends Fragment {

    EditText issue;
    EditText des;
    ImageView imgView;
    private static final int IMAGE_CODE=1;
    Button upload,submit;
    Uri downloadURL;
    StorageReference mstorageReference;
    FirebaseFirestore dbs = FirebaseFirestore.getInstance();
    private final CollectionReference complaintRef = dbs.collection("ComplaintSection");

    Button b1;
    // TODO: Rename parameter arguments, choose names that match
    // the fragment initialization parameters, e.g. ARG_ITEM_NUMBER
    private static final String ARG_PARAM1 = "param1";
    private static final String ARG_PARAM2 = "param2";

    // TODO: Rename and change types of parameters
    private String mParam1;
    private String mParam2;

    public ComplaintFragment() {
        // Required empty public constructor
    }

    /**
     * Use this factory method to create a new instance of
     * this fragment using the provided parameters.
     *
     * @param param1 Parameter 1.
     * @param param2 Parameter 2.
     * @return A new instance of fragment ComplaintFragment.
     */
    // TODO: Rename and change types and number of parameters
    public static ComplaintFragment newInstance(String param1, String param2) {
        ComplaintFragment fragment = new ComplaintFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);

```

```

mParam2 = getArguments().getString(ARG_PARAM2);

}
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
// Inflate the layout for this fragment

View rootView=inflater.inflate(R.layout.fragment_complaint, container, false);
issue=rootView.findViewById(R.id.line_1);
des=rootView.findViewById(R.id.line_2);
upload=rootView.findViewById(R.id.upload_button);

imgView=rootView.findViewById(R.id.rectangle_4);
submit=rootView.findViewById(R.id.file_complaint_button);
b1=rootView.findViewById(R.id.file_complaint_button);

mstorageReference= FirebaseStorage.getInstance().getReference().child("images");
setup();
return rootView;
}

public void setup() {
// TODO define onclicklistener on File Complaints button to collect data from all edittexts and call
ComplaintsUtil.createComplaint
submit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

f(v);

}
});
upload.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent=new Intent().setAction(Intent.ACTION_GET_CONTENT);
intent.setType("image/*");
startActivityForResult(Intent.createChooser(intent, "Complete action using"), IMAGE_CODE);

}
});
// TODO Pass a callback by createComplaint().setOnCompleteListener() to handle the result from the create operation.
Show a toast according to the result and also hide the progress bar.
}
@Override
public void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {

super.onActivityResult(requestCode, resultCode, data);
if (requestCode == IMAGE_CODE && resultCode == Activity.RESULT_OK) {
assert data != null;
Uri selected = data.getData();
try {
Bitmap currentImage = MediaStore.Images.Media.getBitmap
(getActivity().getContentResolver(), selected);

imgView.setImageBitmap(currentImage);
Toast.makeText(getActivity(), "img uploaded", Toast.LENGTH_SHORT).show();

```

```

e.printStackTrace();
}

StorageReference photoRef = mstorageReference.child(selected.getLastPathSegment());
Toast.makeText(getActivity(), "Loading", Toast.LENGTH_SHORT).show();

photoRef.putFile(selected).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
    @Override
    public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
        Task<Uri> urlTask = taskSnapshot.getStorage().getDownloadUrl();
        while (!urlTask.isSuccessful()) ;
        downloadURL = urlTask.getResult();
    }
});
}
}

public void f(View v) {
    String s = String.valueOf(downloadURL);
    String s1 = issue.getText().toString();
    String s2 = des.getText().toString();
    Complaint complaint = new Complaint(s1, s2, s);
    complaintRef.add(complaint);

    Toast.makeText(getActivity(), "complaint added", Toast.LENGTH_SHORT).show();
    b1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            fileComplaint(v);
        }
    });

}

public void fileComplaint(View v)
{
    Intent intent = new Intent(getActivity(), ComplaintActivity.class);
    startActivity(intent);
}

}

```

HomeActivity.java

```

package in.hackslash.messsy.home;

import android.annotation.SuppressLint;
import android.app.Notification;
import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;

```

```

import com.google.android.material.bottomnavigation.BottomNavigationView;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.navigation.NavigationView;
import com.google.firebase.auth.FirebaseAuth;

import in.hackslash.messsy.AbsenceNotification;
import in.hackslash.messsy.R;
import in.hackslash.messsy.complaint.NoticeActivity;
import in.hackslash.messsy.onboarding.LoginActivity;
import in.hackslash.messsy.payment.PaymentDetailsActivity;
import in.hackslash.messsy.vote.VoteActivity;

public class HomeActivity extends AppCompatActivity implements
NavigationView.OnNavigationItemSelectedListener {
    BottomNavigationView bnv;
    private DrawerLayout drawer;
    String name="";
    FloatingActionButton fab;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        bnv = findViewById(R.id.btm_nav);
        //      fab = findViewById(R.id.fab);

        bnv.setBackground(null);
        //      bnv.getMenu().getItem(2).setEnabled(false);

        getSupportFragmentManager().beginTransaction().replace(R.id.frameContainer, new HomeFragment()).commit();
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        drawer = findViewById(R.id.drawer_layout);
        NavigationView navigationView = findViewById(R.id.nav_view);

        navigationView.setNavigationItemSelectedListener(this);
        bnv.setOnNavigationItemSelectedListener(new BottomNavigationView.OnNavigationItemSelectedListener() {
            @SuppressWarnings("NonConstantResourceId")
            @Override
            public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {

                Fragment temp = null;
                switch (item.getItemId()) {
                    case R.id.menu_home:
                        temp = new HomeFragment();
                        break;
                    case R.id.menu_complaint:
                        temp = new ComplaintFragment();
                        break;
                    case R.id.menu_profile:
                        temp = new ProfileFragment();
                        break;

                    //          case R.id.menu_qrCode:
                    //              temp = new QRFragment();
                    //              break;

                }
                getSupportFragmentManager().beginTransaction().replace(R.id.frameContainer, temp).commit();
            }
        });
    }
}

```



```

    }
    });
    // fab.setOnClickListener(new View.OnClickListener() {
    //     @Override
    //     public void onClick(View view) {
    //         startActivity(new Intent(HomeActivity.this, QRScanner.class));
    //     }
    // });
    ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(this, drawer, toolbar,
    R.string.navigation_drawer_open, R.string.navigation_drawer_close);
    drawer.addDrawerListener(toggle);
    toggle.syncState();
}

@Override
public void onBackPressed() {
    if (drawer.isDrawerOpen(GravityCompat.START)) {
        drawer.closeDrawer(GravityCompat.START);
    } else {
        super.onBackPressed();
    }
}

@Override
public boolean onNavigationItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()) {
        case R.id.menu_absence:
            startActivity(new Intent(HomeActivity.this, AbsenceNotification.class));
            break;

        case R.id.menu_notice:
            startActivity(new Intent(HomeActivity.this, NoticeActivity.class));
            break;

        // case R.id.menu_payment:
        //     Toast.makeText(this, "payment done", Toast.LENGTH_SHORT).show();
        //     startActivity(new Intent(this, PaymentDetailsActivity.class));
        //     break;

        case R.id.menu_change:
            startActivity(new Intent(HomeActivity.this, VoteActivity.class));
            break;

        case R.id.logout1:
            FirebaseAuth.getInstance().signOut();
            Toast.makeText(HomeActivity.this, "Logged out!!", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(getApplicationContext(), LoginActivity.class));
            finish();
            break;

        }
        import android.view.ViewGroup;
    return true;
}
}

```

HomeFragment.java

```

package in.hackslash.messsy.home;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;

```



```

mealNext.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Toast.makeText(getActivity(), mealNextType, Toast.LENGTH_SHORT).show();
        fetchMealData(mealNextType, day);
        prev=-1;
    }
});

mealListrecyclerView.addItemTouchListener(new RecyclerViewItemClickListener(getContext(), mealListrecyclerView, new RecyclerViewItemClickListener.OnItemClickListener() {
    @Override public void onItemClick(View view, int position) {
        // do whatever
        if(prev==position)
        {
            return;
        }
        MealData data=mealList.get(position);
        Picasso.with(getContext()).load(data.getImageUrl()).into(foodimg);
        fooddesctv.setText(data.getDescription());
        foodnametv.setText(data.getName());
        mealList.get(position).setCardColor(R.color.main_colour);
        mealList.get(position).setTextColor(R.color.yellow);
        if(prev!=-1)
        {
            mealList.get(prev).setCardColor(R.color.white);
            mealList.get(prev).setTextColor(R.color.black);
        }
        prev=position;
        adapter = new MealListDataAdapter(getContext(), mealList);
        mealListrecyclerView.setAdapter(adapter);
    }

    @Override public void onLongItemClick(View view, int position) {
        // do whatever
    }
}));
mealListrecyclerView.setAdapter(adapter);
fetchMealData(mealType, day);
setupMealSpinner(mealSpinner);
return item;
}

private void setUpMealTime() {
    String currentTime = new SimpleDateFormat("HH:mm:ss", Locale.getDefault()).format(new Date());
    int hours = Integer.parseInt(currentTime.substring(0, 2));
    mealType = (hours > 18 && hours < 24 || hours < 4) ? "dinner" : (hours < 18 && hours > 12) ? "lunch" : "breakfast";
    day = currentDay();

    switch (mealType) {
        case "breakfast":
            mealNextType = "lunch";
            time += "morning";
            break;
        case "lunch":
            mealNextType = "dinner";
            time += "afternoon";
            break;
        case "dinner":
            mealNextType = "lunch"; // should be breakfast
            time += "evening";
    }
}

```

```
break;
}
}
```

```
private void setUpUserName() {
String uid = FirebaseAuth.getInstance().getUid();
FirebaseFirestore.getInstance().collection("users").document(uid).get().
addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
@Override
public void onSuccess(DocumentSnapshot documentSnapshot) {
time += " " +documentSnapshot.getString("name");
name.setText(time);
userName=documentSnapshot.getString("name");
}
}).addOnFailureListener(new OnFailureListener() {
@Override
public void onFailure(@NonNull Exception e) {
Toast.makeText(getContext(), "Something went wrong...", Toast.LENGTH_SHORT).show();
}
});
}
```

```
private void setupView(View item) {
mealListrecyclerView = item.findViewById(R.id.meals_list);
mealListrecyclerView.setLayoutManager(new LinearLayoutManager(getContext(),
LinearLayoutManager.HORIZONTAL, false));
```

```
name=item.findViewById(R.id.name_tv);
image = item.findViewById(R.id.imageView_meal);
mealNext = item.findViewById(R.id.meal_next_button);
mealSpinner = item.findViewById(R.id.spinner);
```

```
foodnametv = item.findViewById(R.id.food_name_tv);
fooddesctv = item.findViewById(R.id.food_description);
foodimg = item.findViewById(R.id.food_img);
}
```

```
private String currentDay(){
Calendar calendar = Calendar.getInstance();
Date date = calendar.getTime();
```

```
String day = new SimpleDateFormat("EEEE", Locale.ENGLISH).format(date.getTime());
return day;
}
```

```
private void fetchMealData(String meal, String day) {
```

```
mealList.clear();
firestore = FirebaseFirestore.getInstance();
firestore.collection("MealListData")
.document(day).collection(meal).get().addOnSuccessListener(new OnSuccessListener<QuerySnapshot>() {
@Override
public void onSuccess(QuerySnapshot queryDocumentSnapshots) {
for (QueryDocumentSnapshot snapshot : queryDocumentSnapshots) {
{
MealData mealData = snapshot.toObject(MealData.class);
if (snapshot.exists()) {
String mealname = mealData.getName();
String description = mealData.getDescription();
String imageUrl = mealData.getImageUrl();
if(prev==1)
{
prev=0;
```

```

Picasso.with(getApplicationContext()).load(mealList.get(0).getImageUrl()).into(foodimg);
fooddesctv.setText(mealList.get(0).getDescription());
foodnametv.setText(mealList.get(0).getName());
continue;
}
mealList.add(new MealData(imageUrl, description, mealname,R.color.white,R.color.black));
}
}
}
adapter = new MealListDataAdapter(getApplicationContext(), mealList);
mealListrecyclerView.setAdapter(adapter);
}
}).addOnFailureListener(new OnFailureListener() {

@Override
public void onFailure(@NonNull Exception e) {
Toast.makeText(getApplicationContext(), "Something went wrong", Toast.LENGTH_SHORT).show();
}
});

}

// @Override
// public void onItemClick(int position) {
//     Log.d(TAG, "Clicked"+position);
//     foodnametv.setText(mealList.get(position).getName());
//     fooddesctv.setText(mealList.get(position).getDescription());
//     Picasso.with(getApplicationContext()).load(mealList.get(position).getImageUrl()).into(foodimg);
// }
private void setupMealSpinner(Spinner mealSpinner) {
ArrayAdapter<String> mealSpinnerAdapter=new ArrayAdapter<String>(getApplicationContext(),
android.R.layout.simple_spinner_item, mealTimeArray);
mealSpinnerAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
mealSpinner.setAdapter(mealSpinnerAdapter);
mealSpinner.setOnItemSelectedListener(new MealSpinner());
}

private class MealSpinner implements AdapterView.OnItemSelectedListener {
@Override
public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l) {
mealSelect= mealTimeArray[i];
String day = currentDay();
fetchMealData(mealSelect,day);
}

@Override
public void onNothingSelected(AdapterView<?> adapterView) {

}
}
}
}

```

```

import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;

import com.bumptech.glide.Glide;
import com.bumptech.glide.request.RequestOptions;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import java.net.URL;

import in.hackslash.messsy.R;
import in.hackslash.messsy.onboarding.EditProfileActivity;
import in.hackslash.messsy.payment.BalanceActivity;

public class ProfileFragment extends Fragment implements View.OnClickListener{

    private ImageView profileImage;
    private TextView name,email,roomNo;
    private Button editProfileBtn, viewCouponBtn;
    private FirebaseFirestore db = FirebaseFirestore.getInstance();
    private FirebaseUser currUser = FirebaseAuth.getInstance().getCurrentUser();

    public ProfileFragment() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View item = inflater.inflate(R.layout.fragment_profile, container, false);
        // Assigning Variables
        profileImage = item.findViewById(R.id.profile_img);
        name = item.findViewById(R.id.name);
        email = item.findViewById(R.id.email);
        roomNo = item.findViewById(R.id.Room_no);
        editProfileBtn = item.findViewById(R.id.edit_profile_btn);
        viewCouponBtn = item.findViewById(R.id.view_coupon_btn);
        //
        editProfileBtn.setOnClickListener(this);
        viewCouponBtn.setOnClickListener(this);
        loadUserDetails();
        return item;
    }

    private void loadUserDetails() {
        if(currUser != null) {
            final String user_id = currUser.getId();

```

```

userDetails.addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
    @Override
    public void onSuccess(DocumentSnapshot documentSnapshot) {
        String uName,uEmail,uRoomNo,urlProfileImage;
        urlProfileImage = documentSnapshot.contains("profileImgUrl") ? documentSnapshot.get("profileImgUrl").toString():
        "";
        if(isValidUrl(urlProfileImage)){
            Glide.with(getActivity())
                .load(urlProfileImage)
                .error(R.mipmap.ic_launcher)
                .apply(RequestOptions.circleCropTransform())
                .into(profileImage);
        }else {
            profileImage.setImageDrawable(getActivity().getDrawable(R.drawable.undraw_profile_pic));
            Toast.makeText(getActivity(),"Profile Pic Not Found",Toast.LENGTH_LONG).show();
        }
        uName = documentSnapshot.contains("name") ? documentSnapshot.get("name").toString(): "NOT FOUND";
        uEmail = documentSnapshot.contains("email") ? documentSnapshot.get("email").toString() : "NOT FOUND" ;
        uRoomNo = "Room No. " + (documentSnapshot.contains("roomno") ? documentSnapshot.get("roomno").toString() :
        "NOT FOUND");
        name.setText(uName);
        email.setText(uEmail);
        roomNo.setText(uRoomNo);
    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull @org.jetbrains.annotations.NotNull Exception e) {
        Log.d("PROFILE_FRAGMENT",e.toString());
    }
});
}else{
    Log.d("PROFILE_FRAGMENT","User Not Found");
}
}

private static boolean isValidUrl(String url){
    /* Try creating a valid URL */
    try {
        new URL(url).toURI();
        return true;
    }
    // If there was an Exception while creating URL object
    catch (Exception e) {
        return false;
    }
}

@Override
public void onClick(View view) {
    switch (view.getId()) {
        case R.id.edit_profile_btn:
            startActivity(new Intent(getActivity(), EditProfileActivity.class));
            break;
        case R.id.view_coupon_btn:
            startActivity(new Intent(getActivity(), BalanceActivity.class));
            break;
        default:
            break;
    }
}
}
}

```

QRFragment.java

```

package in.hackslash.messsy.home;

import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.fragment.app.Fragment;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

import androidmads.library.qrgenearator.QRGContents;
import androidmads.library.qrgenearator.QRGEncoder;
import in.hackslash.messsy.R;

public class QRFragment extends Fragment {
    Long time;
    Button b1;
    StorageReference QRstorageReference;
    String date;
    ImageView img;
    TextView txt;

    public QRFragment() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View rootView = inflater.inflate(R.layout.fragment_q_r, container, false);
        txt = rootView.findViewById(R.id.nameID);
        txt.setText(HomeFragment.userName);
        img = rootView.findViewById(R.id.imageViewQR);
        QRstorageReference = FirebaseStorage.getInstance().getReference().child("QRcode");
        gen();

        return rootView;
    }

    private void gen() {
        final String data = txt.getText().toString();
        FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();
        String email = user.getEmail();
        if (img.getDrawable() == null) {
            QRGEncoder qrgEncoder = new QRGEncoder(email, null, QRGContents.Type.TEXT, 300);
            Bitmap QRBits = qrgEncoder.getBitmap();
            img.setImageBitmap(QRBits);
        }
    }
}

```



```
}
```

QRScanner.java

```
package in.hackslash.messsy.home;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.Manifest;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

import com.budiyev.android.codescanner.CodeScanner;
import com.budiyev.android.codescanner.CodeScannerView;
import com.budiyev.android.codescanner.DecodeCallback;
import com.google.zxing.Result;
import com.karumi.dexter.Dexter;
import com.karumi.dexter.PermissionToken;
import com.karumi.dexter.listener.PermissionDeniedResponse;
import com.karumi.dexter.listener.PermissionGrantedResponse;
import com.karumi.dexter.listener.PermissionRequest;
import com.karumi.dexter.listener.single.PermissionListener;

import in.hackslash.messsy.R;
import in.hackslash.messsy.payment.PaymentMenuActivity;

public class QRScanner extends AppCompatActivity {

    CodeScanner codeScanner;
    CodeScannerView codeScannerView;
    TextView resultdata;
    private String code="messsy";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_q_r_scanner);

        codeScannerView=findViewById(R.id.scanner);
        codeScanner=new CodeScanner(this,codeScannerView);
        resultdata=findViewById(R.id.result);

        codeScanner.setDecodeCallback(new DecodeCallback() {
            @Override
            public void onDecoded(@NonNull final Result result) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {

                        String res=result.getText();
                        if(res.equals(code))
                        {
                            startActivity(new Intent(QRScanner.this, PaymentMenuActivity.class));
                            finish();
                            Toast.makeText(QRScanner.this, "Success", Toast.LENGTH_SHORT).show();
                        }
                        else
                        {
                            Toast.makeText(QRScanner.this, "Failed", Toast.LENGTH_SHORT).show();
                        }
                    }
                });
            }
        });
    }
}
```

```

    }
    });
    }
    });
    codeScannerView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            codeScanner.startPreview();
        }
    });

}

@Override
protected void onResume() {
    super.onResume();

    requestforCamera();

}
private void requestforCamera(){
    Dexter.withActivity(this).withPermission(Manifest.permission.CAMERA).withListener(new PermissionListener() {
        @Override
        public void onPermissionGranted(PermissionGrantedResponse permissionGrantedResponse) {
            codeScanner.startPreview();
        }

        @Override
        public void onPermissionDenied(PermissionDeniedResponse permissionDeniedResponse) {

        }

        @Override
        public void onPermissionRationaleShouldBeShown(PermissionRequest permissionRequest, PermissionToken
        permissionToken) {
            permissionToken.continuePermissionRequest();
        }
    }).check();
}
}
}

```

UpcomingRecycler.java

```

package in.hackslash.messsy.home;

import android.view.View;
import android.view.ViewGroup;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class UpcomingRecycler extends RecyclerView.Adapter<UpcomingRecycler.ViewHolder> {

    // Use upcoming_meal.xml to define the layout of single item of the list and remove this comment when don
    @NonNull
    @Override
    public UpcomingRecycler.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        return null;
    }
}

```

```

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, int position) {

}

@Override
public int getItemCount() {
return 0;
}

public class ViewHolder extends RecyclerView.ViewHolder{

public ViewHolder(@NonNull View itemView) {
super(itemView);
}
} }

```

AccountsUtil.java

```

package in.hackslash.messsy.onboarding;

import android.util.Log;
import android.util.Patterns;

import androidx.annotation.NonNull;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseAuthException;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import java.util.HashMap;

public class AccountsUtil {

private OnCompleteListener onCompleteListener;
FirebaseUser currentUser;

public AccountsUtil createAccount(final User user, String password) {

String email = user.getEmail();
currentUser = FirebaseAuth.getInstance().getCurrentUser();
final FirebaseFirestore firestore = FirebaseFirestore.getInstance();
// Check if a user is logged in and return ALREADY_LOGGED_IN if a logged in user is found
if (currentUser != null) {
onCompleteListener.onComplete(Status.ALREADY_LOGGED_IN, user);
}
// Validate user here and return WRONG_INPUT if anything is wrong
else if (!Patterns.EMAIL_ADDRESS.matcher(email).matches() || password.length() < 6
|| user.getName().length()==0 || user.getRoomno().length()==0) {
onCompleteListener.onComplete(Status.WRONG_INPUT, user);
}
// Check DB if the user exists and return DUPLICATE_USER if user exists

// Create new account using createUserWithEmailAndPassword and return SUCCESS if successful
else {

```

```

.addOnCompleteListener(new OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        if (task.isSuccessful()) {
            //user account created
            currentUser = FirebaseAuth.getInstance().getCurrentUser();
            String userid=currentUser.getId();
            String name=user.getName();
            String roll = user.getRoll();
            String rooomno=user.getRoomno();
            String email=user.getEmail();
            HashMap<String,Object> map=new HashMap<>();
            map.put("name",name);
            map.put("roll",roll);
            map.put("roomno",rooomno);
            map.put("email",email);
            map.put("balance",2000);
            firestore.collection("users").document(userid)
            .set(map).addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    if(task.isSuccessful()){
                        // user data written on DB
                        onCompleteListener.onComplete(Status.SUCCESS, user);
                    }
                }
            });
        } else {
            String error_code = ((FirebaseAuthException) task.getException()).getErrorCode();
            if (error_code.equals("ERROR_EMAIL_ALREADY_IN_USE")) {
                onCompleteListener.onComplete(Status.DUPLICATE_USER, user);
            }
        }
    }
});

// Leave as it is
return this;
}

public AccountsUtil loginUser(final User user, final String password) {

    final String email = user.getEmail();
    FirebaseUser currentUser = FirebaseAuth.getInstance().getCurrentUser();
    final FirebaseFirestore firestore = FirebaseFirestore.getInstance();
    // Check if a user is logged in and return ALREADY_LOGGED_IN if a logged in user is found
    if (currentUser != null) {
        onCompleteListener.onComplete(Status.ALREADY_LOGGED_IN, user);
    }
    // Validate user here and return WRONG_INPUT if anything is wrong
    else if (!Patterns.EMAIL_ADDRESS.matcher(email).matches() || password.length() < 6) {
        onCompleteListener.onComplete(Status.WRONG_INPUT, user);
    }

    // Check DB if the user exists and return USER_DOES_NOT_EXIST if user is not found

    // Login using the credentials and return SUCCESS if successful

    else{
        FirebaseAuth.getInstance().signInWithEmailAndPassword(email,password).addOnCompleteListener(new

```

```

@Override
public void onComplete(@NonNull Task<AuthResult> task) {
    if(task.isSuccessful()){
        // user logged in
        // checking user data in DB
        String user_id=FirebaseAuth.getInstance().getCurrentUser().getUid();
        firestore.collection("users").document(user_id).get().addOnCompleteListener(new
        OnCompleteListener<DocumentSnapshot>() {
            @Override
            public void onComplete(@NonNull Task<DocumentSnapshot> task) {
                if(task.isSuccessful()){
                    DocumentSnapshot doc=task.getResult();
                    if(doc.exists()){
                        //user exist
                        onCompleteListener.onComplete(Status.SUCCESS,user);
                    }
                    else{
                        onCompleteListener.onComplete(Status.USER_DOES_NOT_EXIST,user);
                    }
                }
            }
        });
    }
    else{
        Log.d("AccountsUtil", "onComplete: "+"login failed");
    }
}

// Leave as it is
return this;
}

public AccountsUtil updateUser(final User user, final String password) {

    // TODO Check DB if the user exists and return USER_DOES_NOT_EXIST if user is not found

    // TODO Check if the new details match with the DB and return DUPLICATE_USER if found same

    // TODO Validate user here and return WRONG_INPUT if anything is wrong

    // TODO Update the user data on DB as well as Auth if required and return SUCCESS
    final FirebaseUser currentuser=FirebaseAuth.getInstance().getCurrentUser();
    final String user_id=currentuser.getUid();
    final FirebaseFirestore firestore=FirebaseFirestore.getInstance();
    FirebaseFirestore.getInstance().collection("users").document(user_id).get()
    .addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>() {
        @Override
        public void onComplete(@NonNull Task<DocumentSnapshot> task) {
            if(task.isSuccessful()){
                DocumentSnapshot doc=task.getResult();
                if(doc.exists()){
                    //user exist
                    String name_n=user.getName();
                    String roll_n=user.getRoll();
                    String roomno_n=user.getRoomno();
                    String email_n=user.getEmail();
                    String name_old=doc.getString("name");
                    String roomno_old=doc.getString("roomno");
                    String email_old=currentuser.getEmail();
                    if(roomno_n.length()==0 || name_n.length()==0){

```

```

    }
    else if(roomno_n.equals(roomno_old) || name_n.equals(name_old)){
        onCompleteListener.onComplete(Status.DUPLICATE_USER,user);
    }
    else{
        HashMap<String,Object> map=new HashMap<>();
        map.put("name",name_n);
        map.put("roll",roll_n);
        map.put("roomno",roomno_n);
        firestore.collection("users").document(user_id).update(map);
        // if new email provided then updating it
        if(!email_n.equals(email_old)){
            currentuser.updateEmail(email_n).addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    if(task.isSuccessful()){
                        Log.d("AccountsUtil", "email updated ");
                    }
                }
            });
        }
        if(password.length()>=6){
            currentuser.updatePassword(password).addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    Log.d("AccountsUtil", "password updated");
                }
            });
        }
        onCompleteListener.onComplete(Status.SUCCESS,user);
    }
    else{
        onCompleteListener.onComplete(Status.USER_DOES_NOT_EXIST,user);
    }
    // Leave as it is
    return this;
}

public AccountsUtil fetchUser(final String userID) {

    // TODO Check DB if the user exists and return USER_DOES_NOT_EXIST if user is not found

    // TODO Return the user in a User object if found with a SUCCESS Status

    FirebaseFirestore.getInstance().collection("users").document(userID).get()
        .addOnCompleteListener(new OnCompleteListener<DocumentSnapshot>() {
            @Override
            public void onComplete(@NonNull Task<DocumentSnapshot> task) {
                if(task.isSuccessful()){
                    DocumentSnapshot doc=task.getResult();
                    if(doc.exists()){
                        //user exist
                        User user=doc.toObject(User.class);
                        onCompleteListener.onComplete(Status.SUCCESS,user);
                    }
                    else{
                        User user=new User();

```

```

    }
    }
    });
    return this;
}

public void setOnCompleteListener(AccountsUtil.onCompleteListener onCompleteListener) {
    this.onCompleteListener = onCompleteListener;
}

public interface onCompleteListener {
    void onComplete(Status status, User user);
}

public enum Status {
    SUCCESS, DUPLICATE_USER, WRONG_INPUT, ALREADY_LOGGED_IN, USER_DOES_NOT_EXIST
}
}

```

CreateAccountActivity.java

```

package in.hackslash.messsy.onboarding;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import in.hackslash.messsy.R;
import in.hackslash.messsy.home.HomeActivity;

public class CreateAccountActivity extends AppCompatActivity {
    // Declare instances for all UI elements
    private EditText name_ct, email_ct, room_no_ct, password_ct;
    private EditText roll_ct;
    private Button create_acc;
    private TextView log_in;
    private AccountsUtil accountsUtil;
    private ProgressBar progressBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_create_account);

        assignVariables();
        setup();
    }

    public void assignVariables() {
        // initialize all fields here
        name_ct = findViewById(R.id.name_input);
        email_ct = findViewById(R.id.email_input);

```

```

room_no_ct = findViewById(R.id.room_no_input);
password_ct = findViewById(R.id.password_input);
create_acc = findViewById(R.id.create_acc);
log_in = findViewById(R.id.log_in);
progressBar=findViewById(R.id.signup_progress);
accountsUtil = new AccountsUtil();
}

public void setup(){
handleCreateAccount();
handleLogin();

}

private void handleLogin() {
// define onclicklistener on Sign in button to navigate to LoginActivity and close this activity
log_in.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(CreateAccountActivity.this, LoginActivity.class);
startActivity(intent);
finish();
}
});
}

public void handleCreateAccount() {
// define onclicklistener on Save Details button to collect data from all edittexts and call ComplaintsUtil.createAccount

// Show an indefinite progress bar on the screen as soon as the submit button is clicked to denote that data is being
processed

// Pass a callback by createAccount().setOnCompleteListener() to handle the result from the update operation. Show a
toast according to the result and also hide the progress bar.

// NOTE: If the result is SUCCESS, then close this activity and navigate to HomeActivity
create_acc.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

String name = name_ct.getText().toString();
String email = email_ct.getText().toString();
String roomno = room_no_ct.getText().toString();
String password = password_ct.getText().toString();
String roll = roll_ct.getText().toString();
progressBar.setVisibility(View.VISIBLE);

if(TextUtils.isEmpty(email)){
email_ct.setError("Email is required");
return;
}
if(TextUtils.isEmpty(password)){
password_ct.setError("Password is required");
return;
}
if(TextUtils.isEmpty(roomno)){
room_no_ct.setError("Room No. is required");
return;
}
if(TextUtils.isEmpty(name)){
name_ct.setError("Name is required");
return;
}

```



```

if(TextUtils.isEmpty(roll)){
    name_ct.setError("Roll Number is required");
    return;
}
User user = new User();
user.setName(name);
user.setEmail(email);
user.setRoomno(roomno);
user.setBalance(3000);
user.setRoll(roll);

accountsUtil.createAccount(user, password).setOnCompleteListener(new AccountsUtil.onCompleteListener() {
    @Override
    public void onComplete(AccountsUtil.Status status, User user) {
        if (status == AccountsUtil.Status.ALREADY_LOGGED_IN) {
            //user already logged in
            progressBar.setVisibility(View.INVISIBLE);
            Toast.makeText(CreateAccountActivity.this, "Already Logged in", Toast.LENGTH_SHORT).show();
            Intent intent = new Intent(CreateAccountActivity.this, HomeActivity.class);
            startActivity(intent);
            finish();
        } else if (status == AccountsUtil.Status.WRONG_INPUT) {
            //wrong input
            progressBar.setVisibility(View.INVISIBLE);
            Toast.makeText(CreateAccountActivity.this, "Wrong input", Toast.LENGTH_SHORT).show();
        } else if (status == AccountsUtil.Status.SUCCESS) {
            //account created successfully
            progressBar.setVisibility(View.VISIBLE);
            Toast.makeText(CreateAccountActivity.this, "New account created successfully", Toast.LENGTH_SHORT).show();
            Intent intent = new Intent(CreateAccountActivity.this, HomeActivity.class);
            startActivity(intent);
        } else if (status == AccountsUtil.Status.DUPLICATE_USER) {
            //user already exist
            progressBar.setVisibility(View.INVISIBLE);
            Toast.makeText(CreateAccountActivity.this, "User with this email already exist", Toast.LENGTH_SHORT).show();
        }
    }
});
}
}
}
}
}
}
}

```

EditProfileActivity.java

```

package in.hackslash.messsy.onboarding;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.util.Log;
import android.view.View;
import android.widget.Button;

```

```

import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.bumptech.glide.Glide;
import com.bumptech.glide.request.RequestOptions;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseAuthException;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.SetOptions;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import org.jetbrains.annotations.NotNull;

import java.net.URL;
import java.util.HashMap;
import java.util.Map;

import in.hackslash.messsy.R;
import in.hackslash.messsy.complaint.ComplaintActivity;
import in.hackslash.messsy.complaint.ComplaintsUtil;
import in.hackslash.messsy.home.HomeActivity;

public class EditProfileActivity extends AppCompatActivity {
    public static final int IMAGE_CODE = 1;
    private static final String TAG = "EditProfileActivity";
    private EditText editTextName;
    private EditText editTextRoll;
    private EditText editTextEmail;
    private EditText editTextRoomNo;
    private EditText editTextPassword;
    private Button saveButton, setProfileImg;
    private ImageButton uploadProfileImg;
    private ProgressBar progressBar;
    private ImageView profileImg;
    private FirebaseFirestore db = FirebaseFirestore.getInstance();
    private FirebaseUser currUser = FirebaseAuth.getInstance().getCurrentUser();
    StorageReference mstorageReference;
    Uri profileImgUrl;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit_profile);

        assignVariables();
        showProfileImg();
        progressBar.setVisibility(View.INVISIBLE);
        setup();
    }

```

```

editTextName = findViewById(R.id.name);
editTextRoll = findViewById(R.id.roll);
editTextEmail = findViewById(R.id.email);
editTextRoomNo = findViewById(R.id.room_no);
editTextPassword = findViewById(R.id.password);
saveButton = findViewById(R.id.save_button);
progressBar = findViewById(R.id.progressBar);
profileImg = findViewById(R.id.profile_img);
uploadProfileImag = findViewById(R.id.profile_img_upload_button);
setProfileImg = findViewById(R.id.profile_img_set_button);
mstorageReference= FirebaseStorage.getInstance().getReference().child("images");
}
private void showProfileImg(){
if(currUser != null) {
final String user_id = currUser.getId();
Task<DocumentSnapshot> userDetails = db.collection("users").document(user_id).get();
userDetails.addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
@Override
public void onSuccess(DocumentSnapshot documentSnapshot) {
String imgUrl;
imgUrl = documentSnapshot.contains("profileImgUrl") ? documentSnapshot.get("profileImgUrl").toString(): "";
if(isValidUrl(imgUrl)){
Glide.with(EditProfileActivity.this)
.load(imgUrl)
.error(R.mipmap.ic_launcher)
.apply(RequestOptions.circleCropTransform())
.into(profileImg);
}else {
profileImg.setImageDrawable(getDrawable(R.drawable.undraw_profile_pic));
}
}
}).addOnFailureListener(new OnFailureListener() {
@Override
public void onFailure(@NonNull @org.jetbrains.annotations.NotNull Exception e) {
Log.d("PROFILE_FRAGMENT",e.toString());
}
});
}else{
Log.d("PROFILE_FRAGMENT","User Not Found");
}
}
private static boolean isValidUrl(String url){
/* Try creating a valid URL */
try {
new URL(url).toURI();
return true;
}
// If there was an Exception while creating URL object
catch (Exception e) {
return false;
}
}
public void setup() {
/* NOT REQUIRED TO THESE STEPS BECAUSE TASK IS DONE IN ALTERNATIVE WAY
// define onclicklistener on Save Details button to collect data from all edittexts and call ComplaintsUtil.updateUser
// Show an indefinite progress bar on the screen as soon as the submit button is clicked to denote that data is being
processed

// Pass a callback by updateUser().setOnCompleteListener() to handle the result from the update operation. Show a
toast according to the result and also hide the progress bar.

// NOTE: If the result is SUCCESS, then close this activity and navigate to HomeActivity

```

```

saveButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        progressBar.setVisibility(View.VISIBLE);
        final String name = editTextName.getText().toString();
        final String roll = editTextRoll.getText().toString();
        final String email = editTextEmail.getText().toString();
        final String roomNo = editTextRoomNo.getText().toString();
        final String password = editTextPassword.getText().toString();

        if(name.length() == 0 || roomNo.length() == 0 || email.length() == 0 || roll.length() == 0){
            progressBar.setVisibility(View.GONE);
            Toast.makeText(EditProfileActivity.this, "Empty Fields", Toast.LENGTH_SHORT).show();
            Log.d(TAG, "NOTHING CHANGED");
            return;
        }
        if(password.length() < 6){
            progressBar.setVisibility(View.GONE);
            Toast.makeText(EditProfileActivity.this, "Password's length cannot be less than 6", Toast.LENGTH_SHORT).show();
            Log.d(TAG, "NOTHING CHANGED");
            return;
        }

        final User user = new User(email, name, roomNo, password, 3000, roll);

        if(currUser != null) {
            final String user_id = currUser.getId();
            String oldEmail = currUser.getEmail();
            if(!email.equals(oldEmail)){
                // checking if new email entered is available or not
                currUser.updateEmail(email).addOnCompleteListener(new OnCompleteListener<Void>() {
                    @Override
                    public void onComplete(@NonNull Task<Void> task) {
                        if(task.isSuccessful()){
                            db.collection("users").document(user_id).set(user);
                            progressBar.setVisibility(View.GONE);
                            Log.d(TAG, "EMAIL UPDATED WITH OTHER DETAILS");
                            Toast.makeText(EditProfileActivity.this, "Details Changed", Toast.LENGTH_SHORT).show();
                            intentToHomeActivity();
                        }else{
                            String error_code = ((FirebaseAuthException) task.getException()).getErrorCode();
                            progressBar.setVisibility(View.GONE);
                            if (error_code.equals("ERROR_EMAIL_ALREADY_IN_USE")) {
                                Toast.makeText(EditProfileActivity.this, "Email Already In Use", Toast.LENGTH_SHORT).show();
                            }else{
                                Toast.makeText(EditProfileActivity.this, error_code, Toast.LENGTH_SHORT).show();
                                Log.d(TAG, error_code);
                            }
                        }
                    }
                });
            }else{
                db.collection("users").document(user_id).set(user);
                progressBar.setVisibility(View.GONE);
                Toast.makeText(EditProfileActivity.this, "Details Changed", Toast.LENGTH_SHORT).show();
                intentToHomeActivity();
            }
        }else{
            FirebaseAuth.getInstance().createUserWithEmailAndPassword(email, password)
                .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override

```

```

public void onComplete(@NonNull Task<AuthResult> task) {
    if(task.isSuccessful()){
        db.collection("users").document().set(user);
        progressBar.setVisibility(View.GONE);
        Toast.makeText(EditProfileActivity.this, "New User Created", Toast.LENGTH_SHORT).show();
        intentToHomeActivity();
    }else{
        String error_code = ((FirebaseAuthException) task.getException()).getErrorCode();
        progressBar.setVisibility(View.GONE);
        if (error_code.equals("ERROR_EMAIL_ALREADY_IN_USE")) {
            Toast.makeText(EditProfileActivity.this, "Email Already In Use", Toast.LENGTH_SHORT).show();
        }else{
            Toast.makeText(EditProfileActivity.this, error_code, Toast.LENGTH_SHORT).show();
            Log.d(TAG, error_code);
        }
    }
});
setProfileImg.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        updateImg();
        finish();
    }
});
uploadProfileImg.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent().setAction(Intent.ACTION_GET_CONTENT);
        intent.setType("image/*");
        startActivityResult(Intent.createChooser(intent,"Complete action using"), IMAGE_CODE);
    }
});

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable @org.jetbrains.annotations.Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode == IMAGE_CODE && resultCode == RESULT_OK){
        progressBar.setVisibility(View.VISIBLE);
        assert data != null;
        Uri selected = data.getData();
        StorageReference photoRef = mstorageReference.child(selected.getLastPathSegment());

        photoRef.putFile(selected).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
                Task<Uri> urlTask = taskSnapshot.getStorage().getDownloadUrl();
                while (!urlTask.isSuccessful()) ;
                profileImgUrl = urlTask.getResult();
                String imgUrl = String.valueOf(profileImgUrl);
                if(isValidUrl(imgUrl)){
                    Glide.with(EditProfileActivity.this)
                        .load(imgUrl)
                        .error(R.mipmap.ic_launcher)
                        .apply(RequestOptions.circleCropTransform())
                        .into(profileImg);
                }
            }
        });
    }
}

```

```

progressBar.setVisibility(View.GONE);
}
});
}
}
private void updateImg(){
final String user_id = currUser.getId();
DocumentReference docRef = db.collection("users").document(user_id);
Map<String, Object> map =new HashMap<>();
map.put("profileImgUrl",String.valueOf(profileImgUrl));
docRef.update(map)
.addOnSuccessListener(new OnSuccessListener<Void>() {
@Override
public void onSuccess(Void unused) {
Toast.makeText(EditProfileActivity.this, "Profile Image Updated", Toast.LENGTH_SHORT).show();
}
})
.addOnFailureListener(new OnFailureListener() {
@Override
public void onFailure(@NonNull @NotNull Exception e) {
Log.d(TAG, "onFailure: ",e);
}
});
}
void intentToHomeActivity(){
Intent intent = new Intent(EditProfileActivity.this, HomeActivity.class);
startActivity(intent);
finish();
}
}

```

ForgotPassword.java

```

package in.hackslash.messsy.onboarding;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

import in.hackslash.messsy.R;

public class ForgotPassword extends AppCompatActivity {

    private static final String TAG = "ForgotPassword";
    private EditText email_et;
    private Button btnForgotPass;
    private FirebaseAuth auth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forgot_password);
    }

```

```

email_et = findViewById(R.id.email_edit_text);
btnForgotPass = findViewById(R.id.btn_forgot_pass);
auth = FirebaseAuth.getInstance();

handleForgotPassword();
}

private void handleForgotPassword() {
    btnForgotPass.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            String email = email_et.getText().toString().trim();
            if (TextUtils.isEmpty(email)) {
                Toast.makeText(getApplicationContext(), "Enter your email!", Toast.LENGTH_SHORT).show();
                return;
            }
            sendPasswordResetLink(email);
        }
    });
}

private void sendPasswordResetLink(String email){
    auth.sendPasswordResetEmail(email)
        .addOnCompleteListener(new OnCompleteListener() {
            @Override
            public void onComplete(@NonNull Task task) {
                if (task.isSuccessful()) {
                    Toast.makeText(ForgotPassword.this, "Password Reset link has been sent to the email",
                        Toast.LENGTH_SHORT).show();
                    Log.d(TAG,"Reset link sent to email");
                } else {
                    String exc = task.getException().getMessage();
                    Toast.makeText(ForgotPassword.this, "Invalid Email: " + exc, Toast.LENGTH_SHORT).show();
                    Log.d(TAG,exc);
                }
            }
        });
}
}

```

LoginActivity.java

```

package in.hackslash.messsy.onboarding;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

```

```

import in.hackslash.messsy.R;
import in.hackslash.messsy.home.HomeActivity;

public class LoginActivity extends AppCompatActivity {
// TODO Declare instances for all UI elements
private EditText email_et, password_et;
private Button login;
private TextView create_account, forgot_pass;
private AccountsUtil accountsUtil;
private ProgressBar progressBar;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);

assignVariables();
setup();
}

public void assignVariables() {
// initialize all fields here
email_et = findViewById(R.id.line_1);
password_et = findViewById(R.id.line_2);
login = findViewById(R.id.rectangle_1);
create_account = findViewById(R.id.create_account);
forgot_pass = findViewById(R.id.forgot_password);
progressBar=findViewById(R.id.login_progress);
accountsUtil = new AccountsUtil();
}

public void setup() {
handleLogin();
handleForgotPassword();
handleCreateAccount();
}

private void handleCreateAccount() {
// TODO define onclicklistener on Create Account button to navigate to CreateAccountActivity and close this activity
create_account.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(LoginActivity.this, CreateAccountActivity.class);
startActivity(intent);
finish();
}
});
}

private void handleForgotPassword() {
forgot_pass.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
startActivity(new Intent(LoginActivity.this, ForgotPassword.class));
}
});
}

public void handleLogin() {
// define onclicklistener on Sign In button to collect data from all edittexts and call AccountUtil.loginUser

```


// Show an indefinite progress bar on the screen as soon as the submit button is clicked to denote that data is being processed

// Pass a callback by loginUser().setOnCompleteListener() to handle the result from the update operation. Show a toast according to the result and also hide the progress bar.

// NOTE: If the result is SUCCESS, then close this activity and navigate to HomeActivity

```
login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        progressBar.setVisibility(View.VISIBLE);
        String email = email_et.getText().toString();
        String password = password_et.getText().toString();
        User user = new User();
        user.setEmail(email);
        accountsUtil.loginUser(user, password).setOnCompleteListener(new AccountsUtil.onCompleteListener() {
            @Override
            public void onComplete(AccountsUtil.Status status, User user) {
                if (status == AccountsUtil.Status.ALREADY_LOGGED_IN) {
                    //user already logged in
                    progressBar.setVisibility(View.INVISIBLE);
                    Toast.makeText(LoginActivity.this, "Already Logged in", Toast.LENGTH_SHORT).show();
                } else if (status == AccountsUtil.Status.WRONG_INPUT) {
                    //wrong input
                    progressBar.setVisibility(View.INVISIBLE);
                    Toast.makeText(LoginActivity.this, "wrong input", Toast.LENGTH_SHORT).show();
                } else if (status == AccountsUtil.Status.DUPLICATE_USER) {
                    //user already exist
                    progressBar.setVisibility(View.INVISIBLE);
                    Toast.makeText(LoginActivity.this, "user with this email already exist", Toast.LENGTH_SHORT).show();
                } else if (status == AccountsUtil.Status.SUCCESS) {
                    //logged in successful
                    progressBar.setVisibility(View.INVISIBLE);
                    Toast.makeText(LoginActivity.this, "lo" +
                        "gin successful", Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(LoginActivity.this, HomeActivity.class);
                    startActivity(intent);
                    finish();
                }
            }
        });
    }
});

}

protected void onStart(){
    super.onStart();
    FirebaseUser currentUser = FirebaseAuth.getInstance().getCurrentUser();
    if(currentUser!=null){
        startActivity(new Intent(LoginActivity.this,HomeActivity.class));
        finish();
    }
}
}
```

User.java

```
package in.hackslash.messsy.onboarding;
```

```
public class User {

// TODO add user fields here
private String email,name,roomno,uid,roll;
private int balance;

public User(){

}

public User(String email, String name, String roomno, String uid, int balance, String roll) {
this.email = email;
this.name = name;
this.roomno = roomno;
this.uid=uid;
this.balance=balance;
this.roll = roll;
}

public int getBalance() {
return balance;
}

public void setBalance(int balance) {
this.balance = balance;
}

public String getEmail() {
return email;
}

public void setEmail(String email) {
this.email = email;
}

public String getName() {
return name;
}

public void setName(String name) {
this.name = name;
}

public String getRoomno() {
return roomno;
}

public void setRoomno(String roomno) {
this.roomno = roomno;
}

public String getUid() {
return uid;
}

public void setUid(String uid) {
this.uid = uid;
}

public String getRoll() {
return roll;
}
```

```

}

public void setRoll(String roll) {
this.roll = roll;
}
}

```

BalanceActivity.java

```

package in.hackslash.messsy.payment;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import java.text.NumberFormat;
import java.util.Locale;

import in.hackslash.messsy.R;

public class BalanceActivity extends AppCompatActivity {
private TextView curBalance;
private final FirebaseFirestore db = FirebaseFirestore.getInstance();
private final FirebaseUser currUser = FirebaseAuth.getInstance().getCurrentUser();
private final String user_id = currUser.getId();
private final DocumentReference balRef = db.collection("users")
.document(user_id);

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_balance);

curBalance = findViewById(R.id.balance);
Task<DocumentSnapshot> balDocSnapShot = balRef.get();
balDocSnapShot.addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
@Override
public void onSuccess(DocumentSnapshot documentSnapshot) {
Double number;
if(documentSnapshot.contains("balance")){
number = Double.parseDouble(documentSnapshot.get("balance").toString());
}else {
number = 0.0;
}
String str = NumberFormat.getCurrencyInstance(new Locale("en", "in")).format(number);
curBalance.setText(str);
}
});
}
}
}

```

FinalPayment.java

```

package in.hackslash.messsy.payment;

```

```

import androidx.appcompat.app.AppCompatActivity;

import android.app.AlertDialog;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;

import in.hackslash.messsy.R;
import in.hackslash.messsy.onboarding.User;

public class FinalPaymentActivity extends AppCompatActivity {

    TextView amountView;
    EditText passEdit;
    Button payBtn;
    private int balance;
    private int amount;
    ProgressDialog dialog;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_final_payment);
        setupView();
        dialog.setMessage("Loading Data...");
        dialog.setCancelable(false);
        Intent intent=getIntent();
        amount=intent.getIntExtra("balance",0);
        amountView.setText(amount+"Rs");
        dialog.show();
        FirebaseFirestore.getInstance().collection("users").document(FirebaseAuth.getInstance().getUid())
        .get().addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
            @Override
            public void onSuccess(DocumentSnapshot documentSnapshot) {
                User user=documentSnapshot.toObject(User.class);
                balance=user.getBalance();
                dialog.dismiss();
            }
        });
        payBtn.setOnClickListener(v->
        {
            String roll=passEdit.getText().toString().trim();
            if(roll!=null)
            {
                setupPayment(roll);
            }
        });
    }

    private void setupPayment(String roll) {
        if(balance<amount)
        {
            Toast.makeText(this, "Insufficient Balance!", Toast.LENGTH_SHORT).show();
        }
        else
    }

```

```

dialog.show();
balance=balance-amount;
FirebaseFirestore.getInstance().collection("users").document(FirebaseAuth.getInstance().getUid())
.update("balance",balance).addOnSuccessListener(new OnSuccessListener<Void>() {
@Override
public void onSuccess(Void unused) {
dialog.dismiss();
finish();
}
});
}
}

private void setupView() {
dialog=new ProgressDialog(this);
amoutView=findViewById(R.id.amount_final);
passEdit=findViewById(R.id.payment_final);
payBtn=findViewById(R.id.final_pay);
}
}

```

MenuModel.java

```

package in.hackslash.messsy.payment;

public class MenuModel {
private String name,image;
private int quantity,price;

public MenuModel(String name, String image, int quantity, int price) {
this.name = name;
this.image = image;
this.quantity = quantity;
this.price = price;
}

public MenuModel() {
//needed
}

public String getName() {
return name;
}

public void setName(String name) {
this.name = name;
}

public String getImage() {
return image;
}

public void setImage(String image) {
this.image = image;
}

public int getQuantity() {
return quantity;
}

public void setQuantity(int quantity) {
this.quantity = quantity;
}
}

```

```

public int getPrice() {
    return price;
}

public void setPrice(int price) {
    this.price = price;
}
}

```

PaymentMenuAdapter.java

```

package in.hackslash.messsy.payment;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.google.firebase.Timestamp;

import java.util.ArrayList;
import java.util.Map;

import in.hackslash.messsy.R;

public class PaymentHistoryAdapter extends RecyclerView.Adapter<PaymentHistoryAdapter.ViewHolder> {

    final ArrayList<Map<String, Object>> paymentList;
    public PaymentHistoryAdapter(ArrayList<Map<String, Object>> paymentList) {
        this.paymentList = paymentList;
        notifyDataSetChanged();
    }

    public static class ViewHolder extends RecyclerView.ViewHolder{
        private final TextView date, session, status;
        public ViewHolder(View view) {
            super(view);
            date = view.findViewById(R.id.date);
            session = view.findViewById(R.id.session);
            status = view.findViewById(R.id.status);
        }
        public TextView getDateView(){
            return date;
        }
        public TextView getSessionView(){
            return session;
        }
        public TextView getStatusView(){
            return status;
        }
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext())
            .inflate(R.layout.item_payment_details, parent, false);
    }

```

```
return new ViewHolder(view);
}
```

```
@Override
public void onBindViewHolder(@NonNull ViewHolder viewHolder, int position){
    Timestamp timestamp = (Timestamp) paymentList.get(position).get("DATE");
    viewHolder.getDateView().setText(timestamp.toDate().toString());
    viewHolder.getSessionView().setText(paymentList.get(position).get("SESSION").toString());
    viewHolder.getStatusView().setText(paymentList.get(position).get("STATUS").toString());
}
```

```
@Override
public int getItemCount() {
    return paymentList.size();
}
}
```

Vote.java

```
package in.hackslash.messsy.vote;
```

```
public class Vote {
```

```
    String mealTime, foodItem, addOrRemove;
```

```
    Vote() {}
```

```
    public Vote(String sMealTime, String sFoodItem, String sAddOrRemove) {
        this.mealTime = sMealTime;
        this.foodItem = sFoodItem;
        this.addOrRemove = sAddOrRemove;
    }
```

```
    public String getMealTime() {
        return mealTime;
    }
```

```
    public void setMealTime(String mealTime) {
        this.mealTime = mealTime;
    }
```

```
    public String getFoodItem() {
        return foodItem;
    }
```

```
    public void setFoodItem(String foodItem) {
        this.foodItem = foodItem;
    }
```

```
    public String getAddOrRemove() {
        return addOrRemove;
    }
```

```
    public void setAddOrRemove(String addOrRemove) {
        this.addOrRemove = addOrRemove;
    }
}
```

VoteActivity.java

```
package in.hackslash.messsy.vote;
```

```
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
```

```

import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

import in.hackslash.messsy.AbsenceNotification;
import in.hackslash.messsy.R;
import in.hackslash.messsy.home.HomeActivity;

public class VoteActivity extends AppCompatActivity {

    // TODO Declare instances for all UI elements
    ImageView imageView;

    RadioGroup radioGroup;
    RadioButton radioButton;

    EditText FoodItem;

    String addOrRemove = "add";

    FirebaseFirestore firebaseFirestore = FirebaseFirestore.getInstance();
    DocumentReference documentReference;
    FirebaseAuth firebaseAuth ;
    FirebaseUser currentUser;
    FirebaseStorage storage = FirebaseStorage.getInstance();

    Bitmap bitmap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_vote);
        assignVariables();
        setup();
        imageView=findViewById(R.id.arrow_3);
        imageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

```



```

}

});

firebaseAuth = FirebaseAuth.getInstance();
currentUser = firebaseAuth.getCurrentUser();
documentReference = FirebaseFirestore.getInstance().collection("Menu Change").document();

radioGroup = findViewById(R.id.radio_bottom_group);
FoodItem = findViewById(R.id.enter_food_item);

Button add = findViewById(R.id.vote_for_ad);
Button remove = findViewById(R.id.vote_for_re);

Button submit = findViewById(R.id.submit_button);

add.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        addOrRemove = "add";
    }
});

remove.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        addOrRemove = "remove";
    }
});

submit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String food = FoodItem.getText().toString();

        if(food.isEmpty())
        {
            FoodItem.setError("Specify Food Item.");
            return;
        }

        int radioId = radioGroup.getCheckedRadioButtonId();
        radioButton = findViewById(radioId);

        String foodTime = radioButton.getText().toString();

        Vote voteForChange = new Vote(foodTime, food, addOrRemove);

        firestore.collection("Menu Change").document().set(voteForChange).addOnSuccessListener(new
        OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {

                Toast.makeText(VoteActivity.this, "Request sent to " + addOrRemove + " " + food + " in " + foodTime + ".",
                Toast.LENGTH_SHORT).show();

            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {

```

```
Toast.makeText(VoteActivity.this, "Request Failed.", Toast.LENGTH_SHORT).show();
```

```
}
});
```

```
}
});
}
```

```
public void assignVariables() {
// TODO initialize all fields here
}
```

```
public void setup() {
// TODO define onclicklistener on File User button to collect data from all edittexts and call VoteUtil.createVote
```

```
// TODO Show an indefinite progress bar on the screen as soon as the submit button is clicked to denote that data is
being processed
```

```
// TODO Pass a callback by createVote().setOnCompleteListener() to handle the result from the create operation.
Show a toast according to the result and also hide the progress bar.
}
}
```

VoteUtil.java

```
package in.hackslash.messsy.vote;
```

```
public class VoteUtil {
```

```
private onCompleteListener onCompleteListener;
```

```
public VoteUtil createVote(Vote vote){
```

```
// TODO Check if a user is logged in and return NO_USER if no logged in user is found
```

```
// TODO Validate vote here and return WRONG_INPUT if anything is wrong
```

```
// TODO Check DB if the vote exists and return DUPLICATE_COMPLAINT if user exists
```

```
// TODO Create new user using FirebaseDatabase.getInstance().getReference().child('votes/' +
vote.getMealType).push().set(vote) and return SUCCESS if successful
```

```
// Leave as it is
return this;
}
```

```
public void setOnCompleteListener(VoteUtil.onCompleteListener onCompleteListener) {
this.onCompleteListener = onCompleteListener;
}
```

```
public interface onCompleteListener {
void onComplete(Status status, Vote vote);
}
```

```
public enum Status {
SUCCESS, DUPLICATE_VOTE, WRONG_INPUT, NO_USER
}
}
```

```
package in.hackslash.messsy;
```

```

public class AbsenceModelClass {

String startdate, enddate, name, roll, hostel;

AbsenceModelClass(){ }

public AbsenceModelClass(String startdate, String enddate, String name, String roll, String hostel){
this.startdate=startdate;
this.enddate=enddate;
this.name=name;
this.roll=roll;
this.hostel=hostel;
}

public String getStartdate() {
return startdate;
}

public void setStartdate(String startdate) {
this.startdate = startdate;
}

public String getEnddate() {
return enddate;
}

public void setEnddate(String enddate) {
this.enddate = enddate;
}

public String getName() {
return name;
}

public void setName(String name) {
this.name = name;
}

public String getRoll() {
return roll;
}

public void setRoll(String roll) {
this.roll = roll;
}

public String getHostel() {
return hostel;
}

public void setHostel(String hostel) {
this.hostel = hostel;
}
}

```

AbsenceNotification.java

```

package in.hackslash.messsy;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

```

```

import android.content.Intent;
import android.os.Bundle;
import android.print.PrinterId;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

import java.util.HashMap;
import java.util.Map;

import in.hackslash.messsy.home.HomeActivity;
import in.hackslash.messsy.onboarding.User;

public class AbsenceNotification extends AppCompatActivity {

    String START_DATE="START DATE";
    String END_DATE="END DATE";
    String hostel = "Brahmputra";
    ImageView imageView;
    EditText startDate;
    EditText endDate;
    Button okbtn,cancelbtn;

    String sName, sRoll;

    FirebaseFirestore firebaseFirestore = FirebaseFirestore.getInstance();
    DocumentReference documentReference, documentReference2;
    FirebaseAuth firebaseAuth ;
    FirebaseUser currentUser;
    FirebaseStorage storage = FirebaseStorage.getInstance();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_absence_notification);
        startDate = findViewById(R.id.start_date_input);
        endDate = findViewById(R.id.end_date_input);
        okbtn = findViewById(R.id.OK);
        cancelbtn = findViewById(R.id.cancel);
        imageView=findViewById(R.id.arrow_2);

        firebaseAuth = FirebaseAuth.getInstance();
        currentUser = firebaseAuth.getCurrentUser();
        documentReference = FirebaseFirestore.getInstance().collection("AbsenceNotification").document();
        documentReference2 = FirebaseFirestore.getInstance().collection("users").document(currentUser.getId());

```

```

imageView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        startActivity(new Intent(AbsenceNotification.this, HomeActivity.class));
    }
});

```

```

documentReference2.get().addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
    @Override
    public void onSuccess(DocumentSnapshot documentSnapshot) {
        if (documentSnapshot.exists())
        {
            User user = documentSnapshot.toObject(User.class);
            sName = user.getName();
            sRoll = user.getRoll();
        }

        else
        {
            sName = "Unnown";
            sRoll = "Unknown";
            Toast.makeText(AbsenceNotification.this, "Something went wrong!!", Toast.LENGTH_SHORT).show();
        }
    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        sName = "Unnown";
        sRoll = "Unknown";
        Toast.makeText(AbsenceNotification.this, "Something went wrong!!", Toast.LENGTH_SHORT).show();
    }
});

```

```

okbtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v)
    {
        START_DATE = startDate.getText().toString();
        END_DATE = endDate.getText().toString();

```

```

        if(START_DATE.isEmpty())
        {
            startDate.setError("Date Required!");
            return;
        }

```

```

        if(END_DATE.isEmpty())
        {
            endDate.setError("Date Required!");
            return;
        }

```

```

AbsenceModelClass data = new AbsenceModelClass(START_DATE,END_DATE,sName,sRoll,hostel);

```

```

firebaseFirestore.collection("AbsenceNotification").document().set(data).addOnSuccessListener(new
OnSuccessListener<Void>() {
    @Override
    public void onSuccess(Void aVoid) {

```

```

        Toast.makeText(AbsenceNotification.this, "Done.", Toast.LENGTH_SHORT).show();
    }
}

```

```

}
}).addOnFailureListener(new OnFailureListener() {
@Override
public void onFailure(@NonNull Exception e) {

Toast.makeText(AbsenceNotification.this, e.getMessage(), Toast.LENGTH_SHORT).show();

}
});
}
});

}

}

```

OnBoardingActivity.java

```

package in.hackslash.messsy;

import android.content.Intent;
import android.os.Bundle;
import android.text.Html;
import android.view.View;
import android.view.WindowManager;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import in.hackslash.messsy.onboarding.LoginActivity;

public class OnBoardingActivity extends AppCompatActivity {

    ViewPager viewPager;
    LinearLayout dotsLayout;

    SliderAdapter sliderAdapter;
    TextView[] dots;

    int currentPosition;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_on_boarding);
        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

        viewPager = findViewById(R.id.slider);
        dotsLayout = findViewById(R.id.dots);

        sliderAdapter = new SliderAdapter(this);

        viewPager.setAdapter(sliderAdapter);

        addDots(0);
        viewPager.addOnPageChangeListener(changeListener);
    }

    public void next(View view)

```

```

if(currentPosition == 2){
    skip(view);
}

viewPager.setCurrentItem(currentPosition + 1);

}

public void skip(View view) {
    startActivity(new Intent(getApplicationContext(), LoginActivity.class));
    fileList();
    finish();
}

private void addDots(int position){

    dots = new TextView[3];
    dotsLayout.removeAllViews();

    for(int i=0; i<dots.length; i++){

        dots[i] = new TextView(this);
        dots[i].setText(Html.fromHtml("&#1470;"));
        dots[i].setTextSize(100);

        dotsLayout.addView(dots[i]);

    }

    if(dots.length>0){
        dots[position].setTextColor(getResources().getColor(R.color.colorPrimaryDark));
    }

}

ViewPager.OnPageChangeListener changeListener = new ViewPager.OnPageChangeListener() {
    @Override
    public void onPageScrolled(int position, float positionOffset, int positionOffsetPixels) {

    }

    @Override
    public void onPageSelected(int position) {
        addDots(position);
        currentPosition = position;
    }

    @Override
    public void onPageScrollStateChanged(int state) {

    }
};

}

```

SliderAdapter.java

```
package in.hackslash.messsy;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.constraintlayout.widget.ConstraintLayout;
import androidx.viewpager.widget.PagerAdapter;

public class SliderAdapter extends PagerAdapter {

    Context context;
    LayoutInflater layoutInflater;

    public SliderAdapter(Context context) {
        this.context = context;
    }

    int images[] = {
        R.drawable.slider_fooding_made_easier,
        R.drawable.slider_personalized_experience,
        R.drawable.slider_be_informed
    };

    int headings[] = {
        R.string.first_slide_title,
        R.string.second_slide_title,
        R.string.third_slide_title
    };

    int descriptions[] = {
        R.string.first_slide_desc,
        R.string.second_slide_desc,
        R.string.third_slide_desc
    };

    @Override
    public int getCount() {
        return headings.length;
    }

    @Override
    public boolean isViewFromObject(@NonNull View view, @NonNull Object object) {
        return view == (ConstraintLayout) object;
    }

    @NonNull
    @Override
    public Object instantiateItem(@NonNull ViewGroup container, int position) {

        layoutInflater = (LayoutInflater) context.getSystemService(context.LAYOUT_INFLATER_SERVICE);
        View view = layoutInflater.inflate(R.layout.slides_layout, container, false);

        ImageView imageView = view.findViewById(R.id.slider_image);
        TextView heading = view.findViewById(R.id.slider_heading);
        TextView desc = view.findViewById(R.id.slider_description);
```



```

imageView.setImageResource(images[position]);
heading.setText(headings[position]);
desc.setText(descriptions[position]);

container.addView(view);

return view;
}

@Override
public void destroyItem(@NonNull ViewGroup container, int position, @NonNull Object object) {
    container.removeView((ConstraintLayout)object);
}
}

```

Splash_Screen.java

```

package in.hackslash.messsy;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ImageView;

import in.hackslash.messsy.home.HomeActivity;
import in.hackslash.messsy.onboarding.LoginActivity;

public class Splash_screen extends AppCompatActivity {

    SharedPreferences onBoardingScreen;
    private ImageView iv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash_screen);
        iv=findViewById(R.id.logo);
        Animation anim = AnimationUtils.loadAnimation(this,R.anim.transition);
        iv.startAnimation(anim);

        Thread timer = new Thread(){
            public void run() {
                try{
                    sleep(3000);
                }
                catch (InterruptedException e){
                    e.printStackTrace();
                }
                finally {
                    onBoardingScreen = getSharedPreferences("onBoardingScreen", MODE_PRIVATE);
                    boolean isFirstTime = onBoardingScreen.getBoolean("firstTime", true);

                    if(isFirstTime){

                        SharedPreferences.Editor editor = onBoardingScreen.edit();
                        editor.putBoolean("firstTime", false);
                        editor.commit();
                    }
                }
            }
        };
        timer.start();
    }
}

```

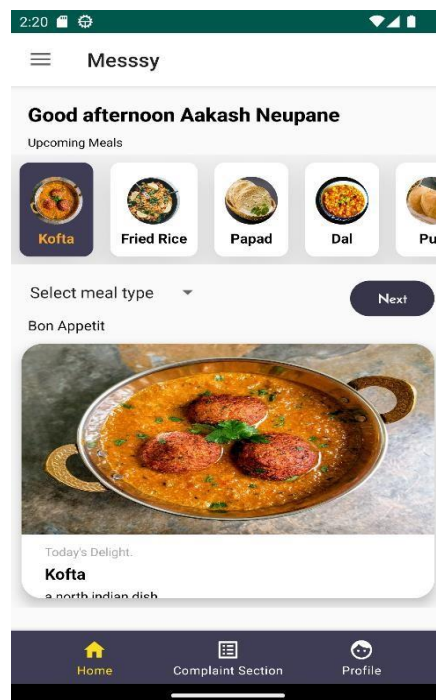
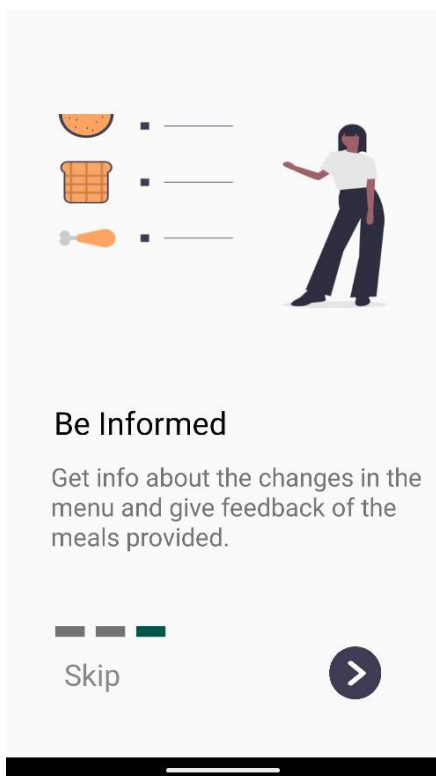
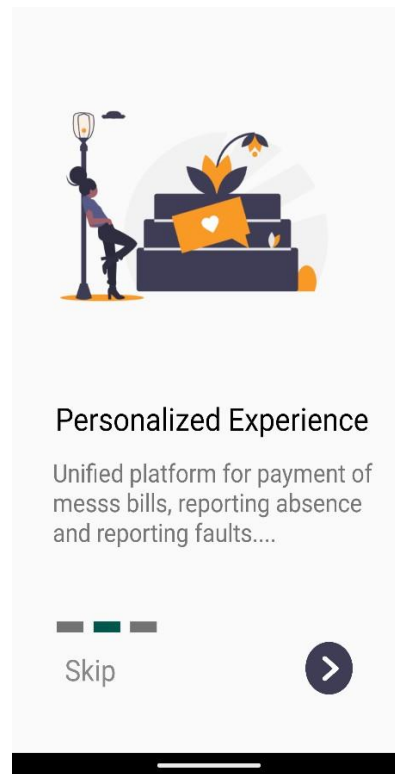
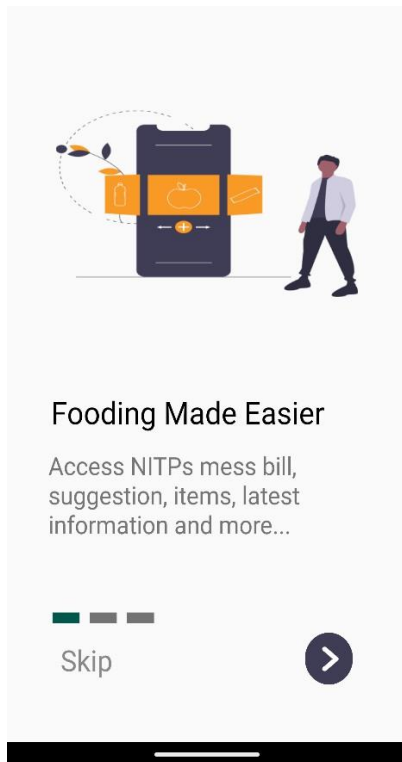
```
Intent i=new Intent(getApplicationContext(), OnBoardingActivity.class);
startActivity(i);
finish();

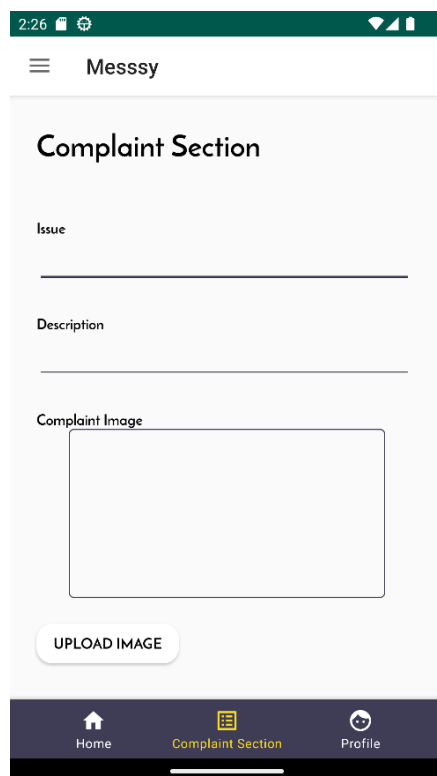
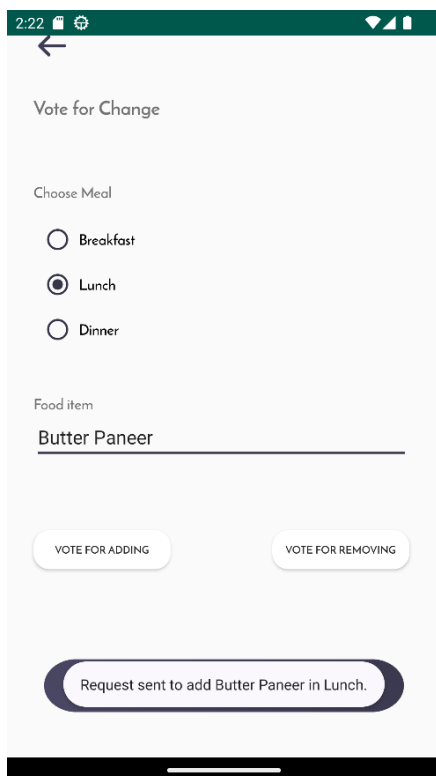
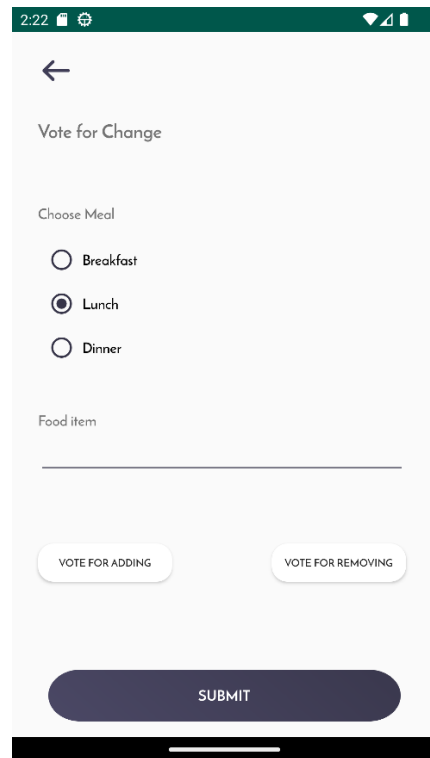
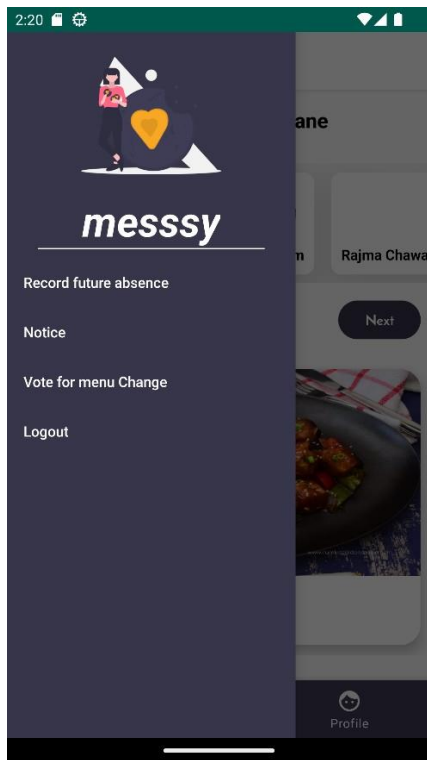
}

else{
Intent i=new Intent(getApplicationContext(), LoginActivity.class);
startActivity(i);
finish();
}

}
}
};
timer.start();
}
}
```


8.1 Result/Output





2:26

Edit your details



Name

Roll Number

Email Address

Room Number

Enter a password



CHAPTER IX

CONCLUSION

Our project “HOSTEL MANAGEMENT SYSTEM” is a very helpful and important project that will manage various activities in the hostel like accommodation, rents, student records and many other things that are very useful for a well- managed hostel. The scope of this project is to provide the facility of the living to the student of rural areas, who are unable to study further in their area. They will be able to have an affordable rate of living in hostel and they will be fully provided the environment which they need for their studies. Another benefit of this project is that all the hostel managing works can be done easily through it by saving our time and also by saving the human efforts. This management system will be an errorless or bugs free management system and will be able to show the records of even years without any confliction.

REFERENCE**CHAPTER X**

- ✓ <https://www.youtube.com/watch?v=RvboTVPRpUI>
- ✓ <https://www.slideshare.net/farhanamusthafa/online-hostel-management-system>
- ✓ <https://github.com/NishkarshRaj/Hostel-Pass-Management-Application.git>
- ✓ <https://www.w3schools.com> › java