

Mobile Application Programming

A Project Report

On

Book Exchange Application

Towards Partial Fulfilment Of

Master of Computer Application (MCA-2)

2020-21



Maharaja Sayajirao University, Vadodara

DEVELOPED BY

PROJECT GUIDE

Ms. Kanika Lakhani

Name	Roll No.
Shivam Pandya	19

Synopsis of Book Exchange System

Modules:

1. Login Form

- The login form consist of email, password.
- Forgot password feature will be present below the sign in details

2. Registration Form

- The registration form consist of name, address, mobile no, email or facebook details and password and confirm password option.

3. Main Page

- The main page will show all the books that are for exchange and the books which are for sale.

4. Sidebar

- Sidebar will consist of :
 - Profile : which will help user to update his/her profile.
 - Add Book : Here the user will add the books.
 - My Books : Here the user will be shown all the books he/she uploaded.
 - Category : We will be bifurcating this page by the category of that book.
 - Logout : The User can leave the application from here.

How will a user will be exchanging or purchasing the book?

- If a user likes a book from another user that he/she wants to exchange or purchase.
- He/she will click on the book they liked which will redirect the user to the book and the person who have uploaded the book and below that the call option will be given.

So when the user will press the chat button, the user will be redirected to the dialer where the owner's phone number is already entered, from where the user can call the owner and discuss about the book.

Brief Introduction of Bookoholic Application

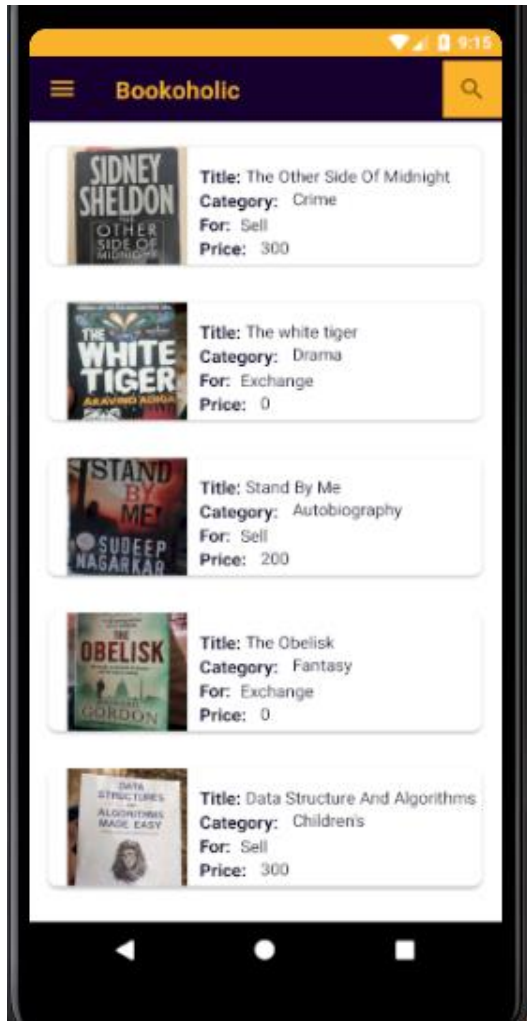
This application is created because there is no such application in the market. And the person who is interested in exchanging the book might not know someone who is interested in the same. So through this platform a community will be created who likes reading books and exchanging them.

The objective of this project is to create and implement an Application for online selling, purchasing and exchanging paperback books. The Application will be used by all sorts of book readers who want to sell, exchange or purchase books at a cheap price. The Application will also allow users to create and maintain individual secured accounts, select their choices, search books in the database. Users can also update their profile and can update or delete the book they put on the application. Users will also be able to contact Application administrators.

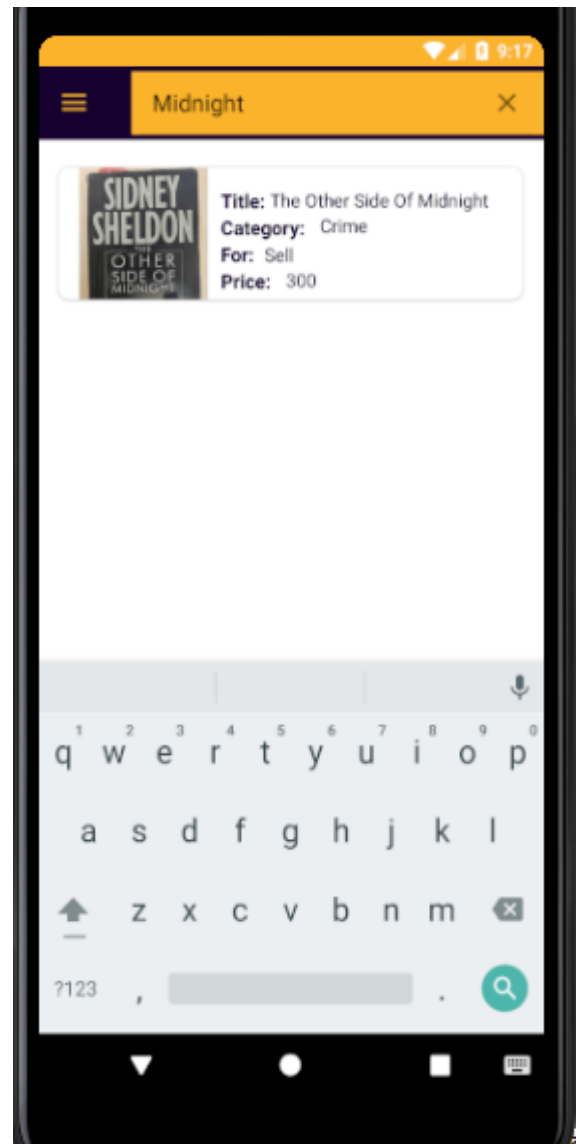
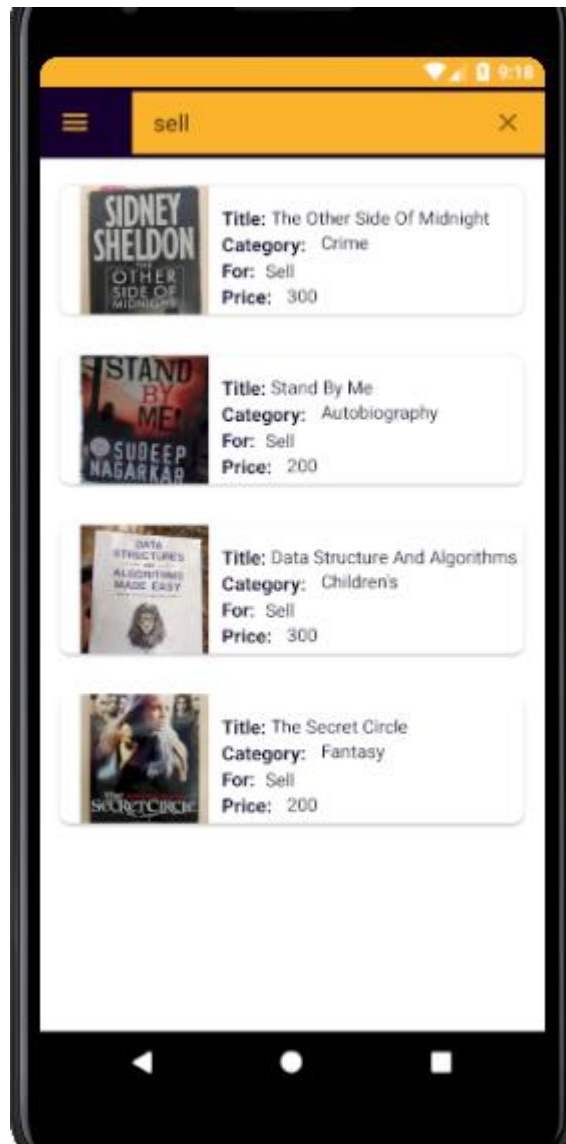
Splash screen: This page is the very first page of the application with a bit of animation.



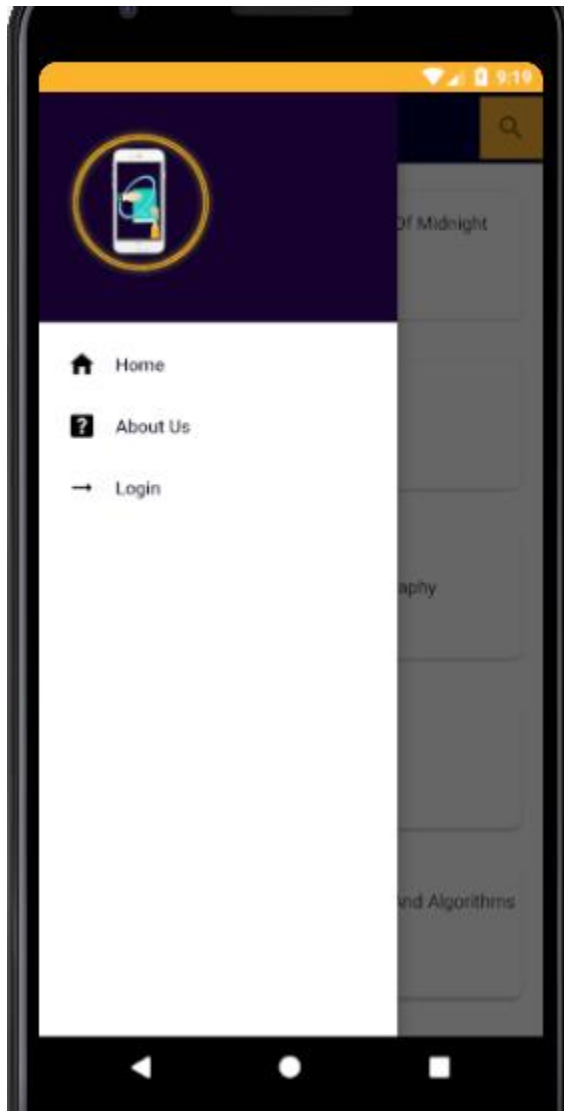
Dashboard: When new user opens this application for the first time they will see the dashboard.



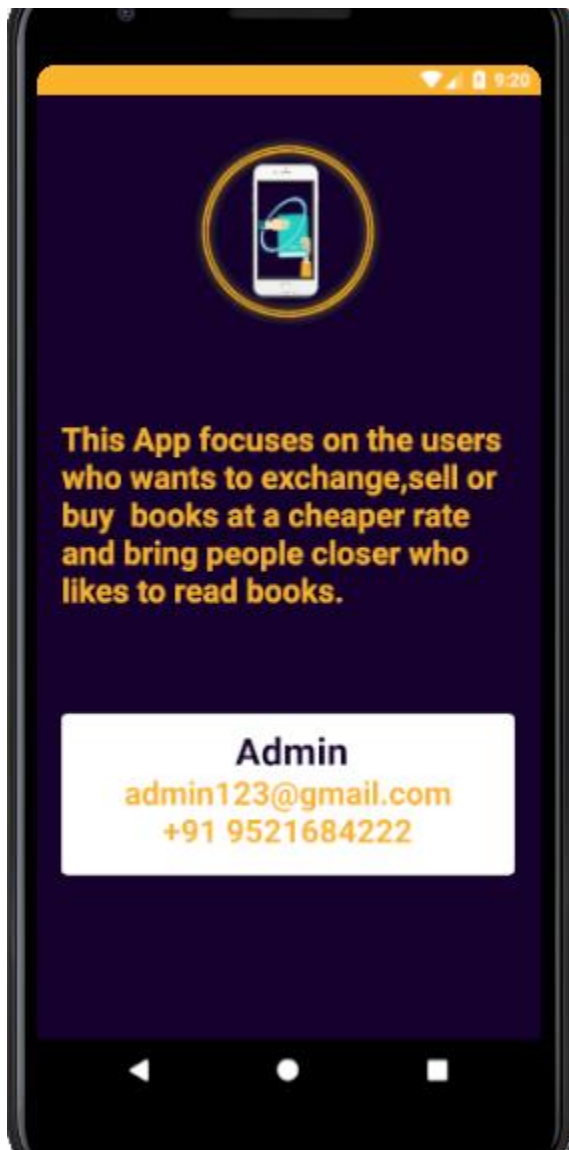
Search option: Users can search books by its title or by if for sell or exchange.



Sidebar: Sidebar for new user. User can see the About Us page and can perform Login operation and clicking on home will redirect the user to the original page i.e. dashboard.



About Us: This page provides the basic information about this application.




Login: In the Login Screen user can Login using credentials, forget password or if user is new and don't have an account then user can make a new account by clicking on NEW USER? SIGN UP.



The image shows a mobile application login screen for 'Bookoholic'. At the top, there is a circular logo featuring a smartphone with a book icon on its screen. Below the logo, the brand name 'Bookoholic' is written in a cursive script. The login form consists of two input fields: 'Phone Number' and 'Password'. The 'Password' field includes a toggle icon for visibility. Below these fields is a link for 'FORGET PASSWORD?'. At the bottom of the form are two buttons: a dark blue button labeled 'GO' and a red button labeled 'NEW USER? SIGN UP'. The entire screen is framed by a black border, and the bottom of the image shows the standard Android navigation bar.


Bookoholic

Phone Number

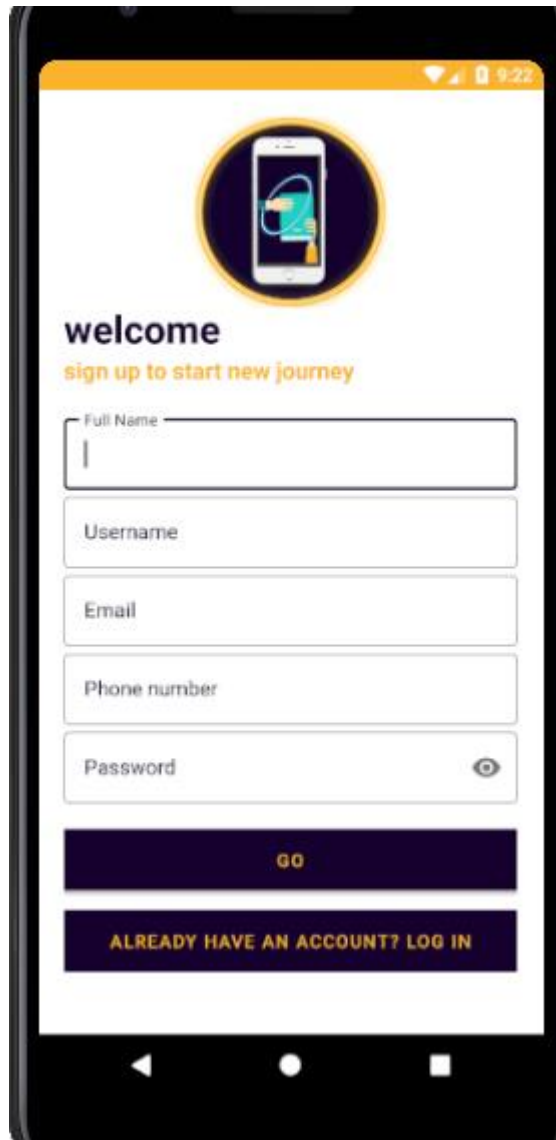
Password 

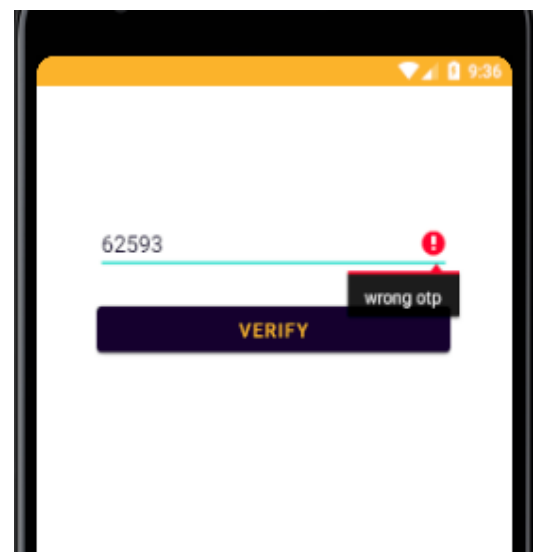
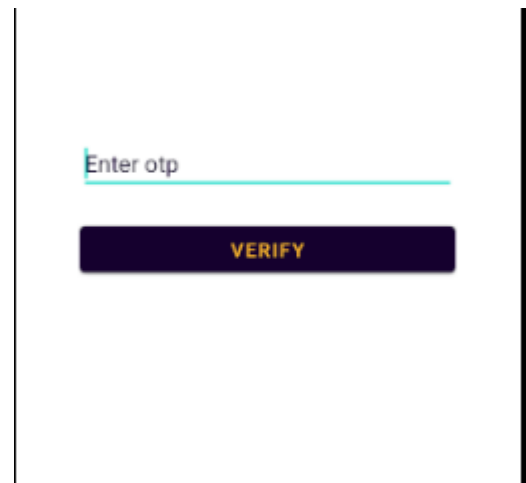
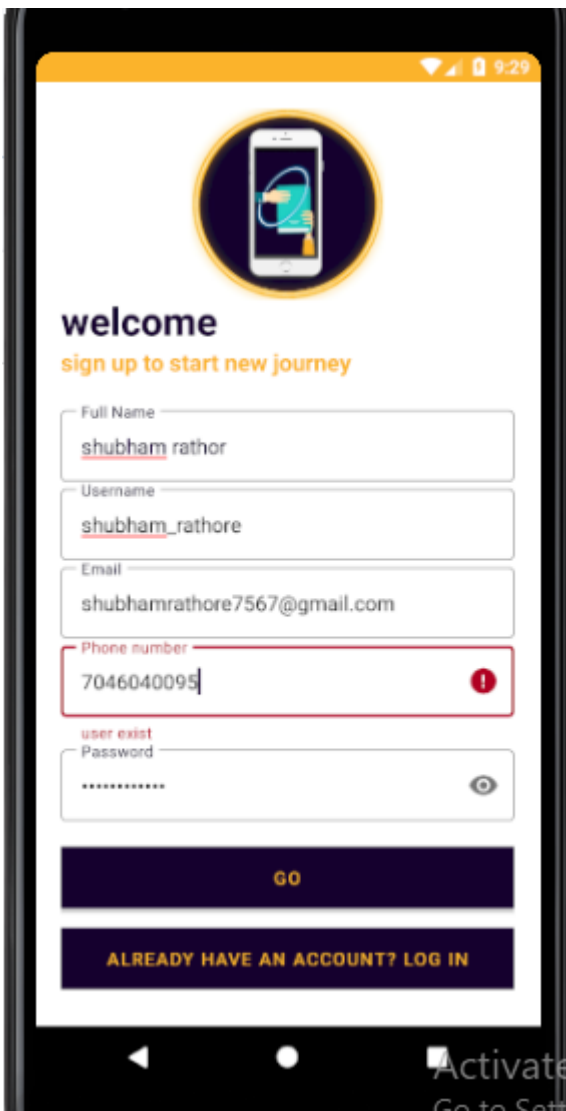
[FORGET PASSWORD?](#)

[GO](#)

[NEW USER? SIGN UP](#)

Registration: Users will fill their details in the form to register. If a user already exists he/she can not register again. It will show a toast “User is already Registered”. User is exists or not will be checked using a phone number.

A screenshot of a mobile application's registration screen. At the top, there is a yellow status bar with a signal strength icon, a battery icon, and the time 9:22. Below this is a circular logo featuring a smartphone with a blue and green graphic on its screen. The text "welcome" is displayed in a bold, black font, followed by "sign up to start new journey" in a smaller, orange font. The registration form consists of five input fields: "Full Name" (with a cursor), "Username", "Email", "Phone number", and "Password" (which includes a toggle icon for visibility). Below the form are two buttons: a dark blue button with the text "GO" in orange, and a dark blue button with the text "ALREADY HAVE AN ACCOUNT? LOG IN" in orange. The bottom of the screen shows a black navigation bar with three white icons: a back arrow, a circle, and a square.



If the user does not exist it'll send an OTP to your mobile number and redirect to OTP page where the verification will be done. If the user enters wrong OTP it'll not proceed. If the user enters right OTP it'll proceed and redirect the page on Login.

Login: If user enters wrong password it'll not proceed else it'll redirect on Dashboard.



The login screen for the Bookoholic app. At the top, there is a circular logo featuring a smartphone with a book icon. Below the logo, the word "Bookoholic" is written in a cursive font. The login form consists of two input fields: "Phone Number" with the value "7434848991" and "Password" with masked characters "*****". A red exclamation mark icon is visible next to the password field, and the text "wrong password" is displayed below it. Below the password field, there is a link "FORGET PASSWORD?". At the bottom, there are two buttons: "GO" and "NEW USER? SIGN UP".

Phone Number
7434848991

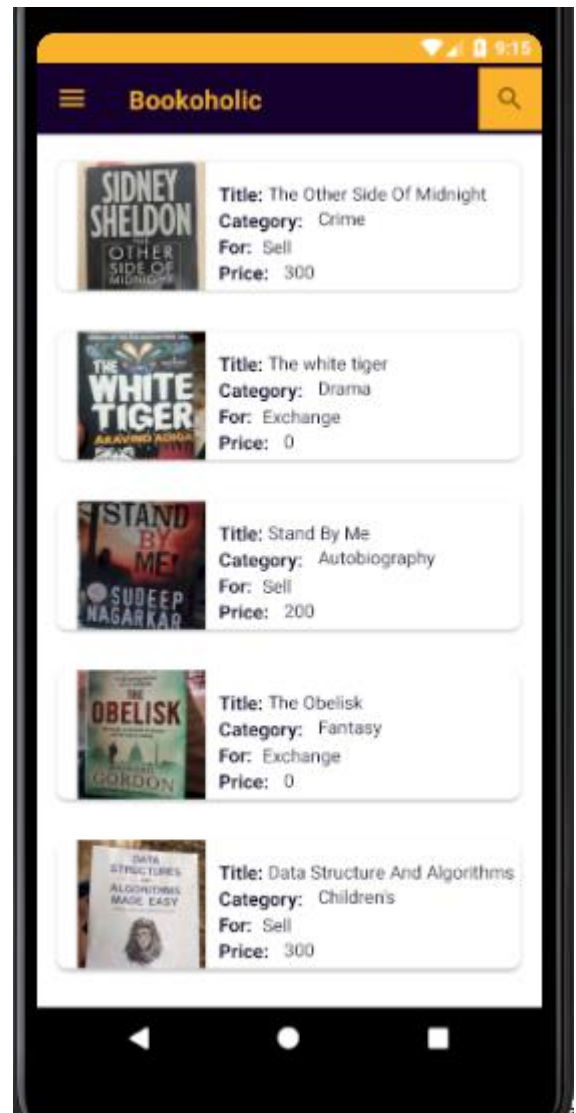
Password

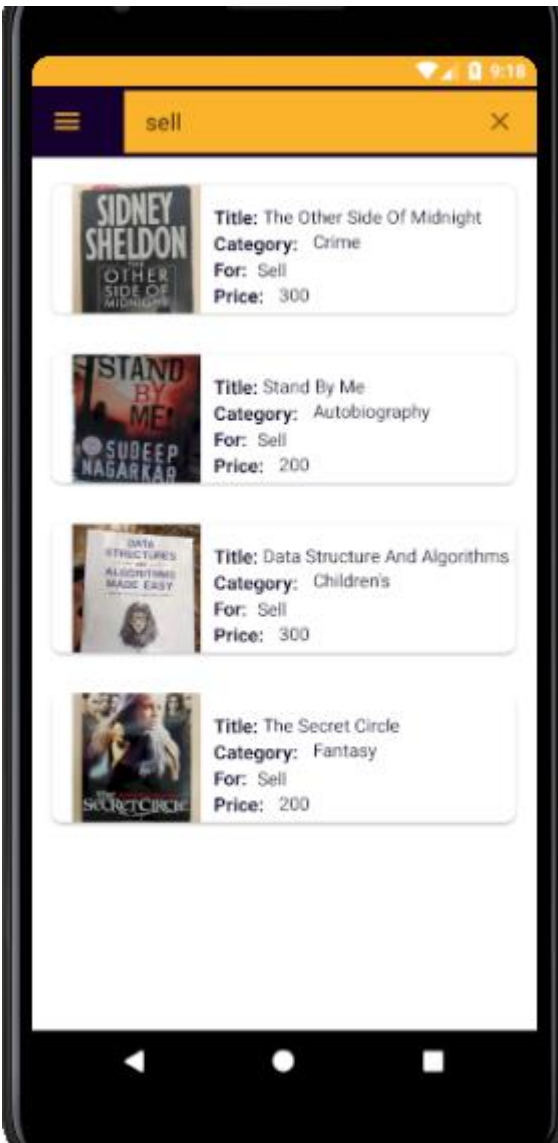
wrong password

FORGET PASSWORD?

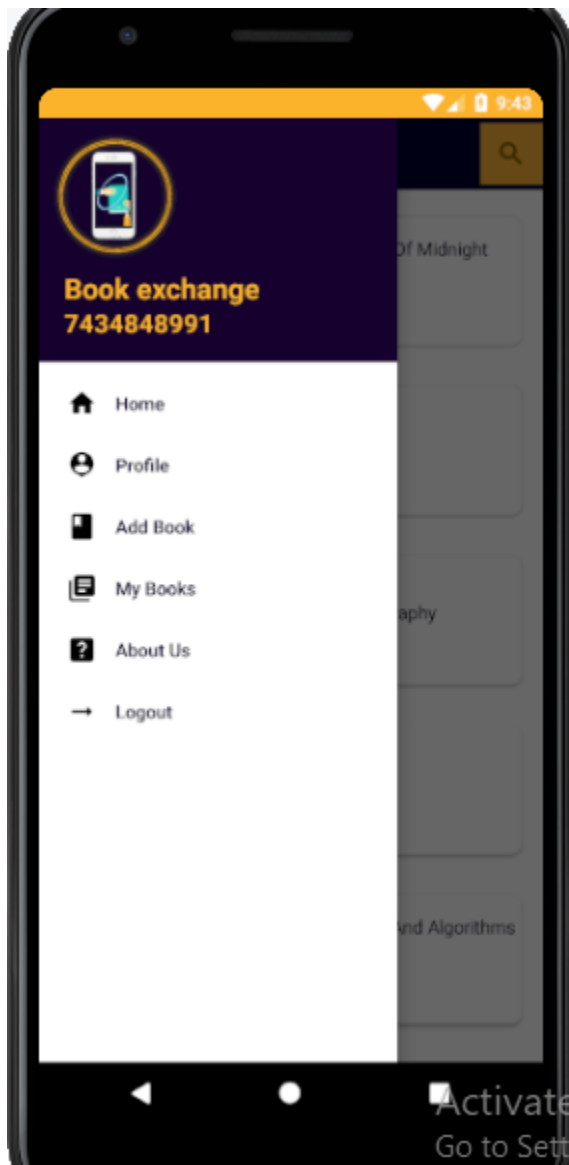
GO

NEW USER? SIGN UP

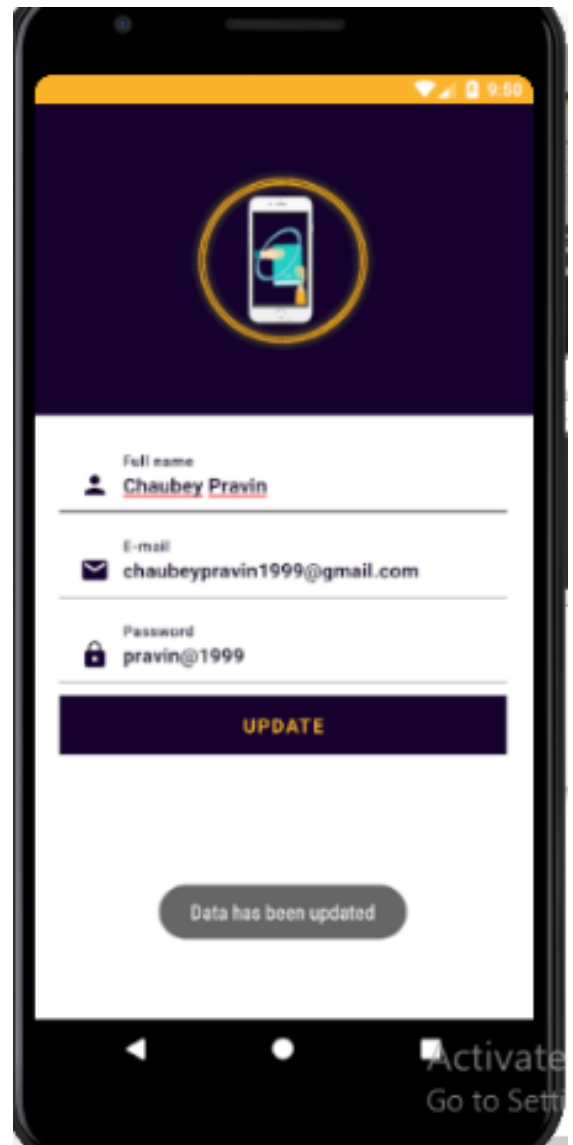
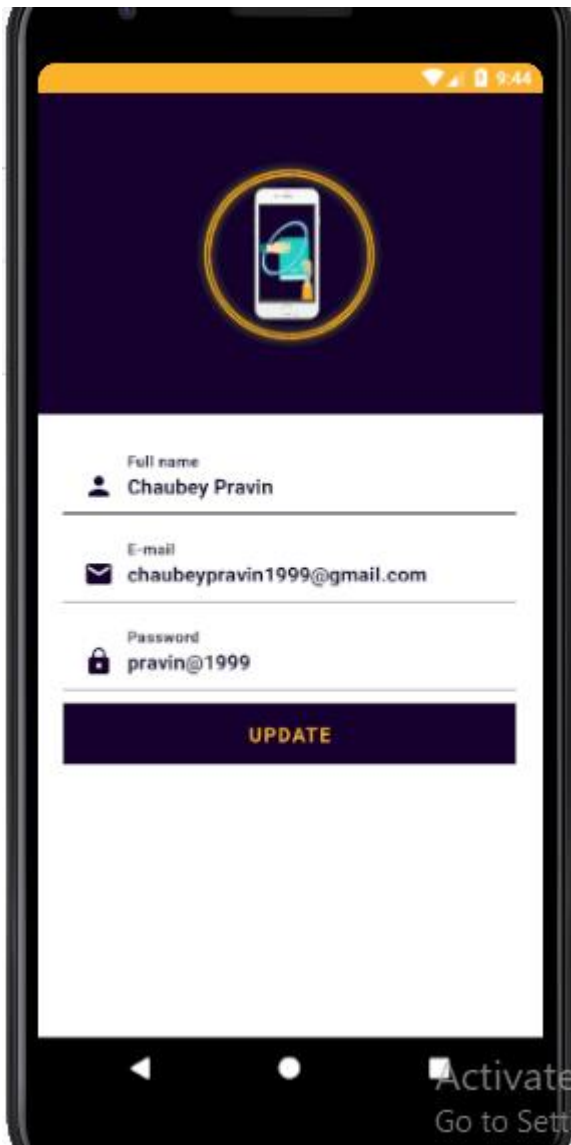


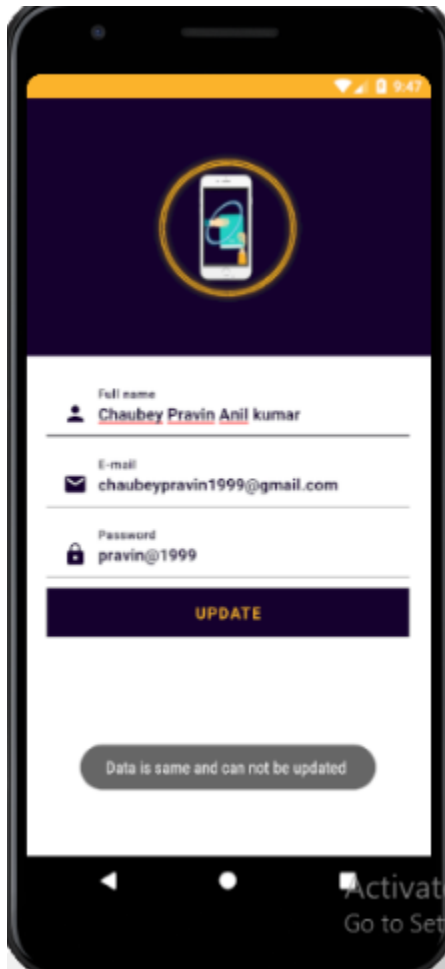


Sidebar after Login: After the user logs into the system he/she will be getting many more options such as updating there profile, adding new books, removing the books you have uploaded and many more.

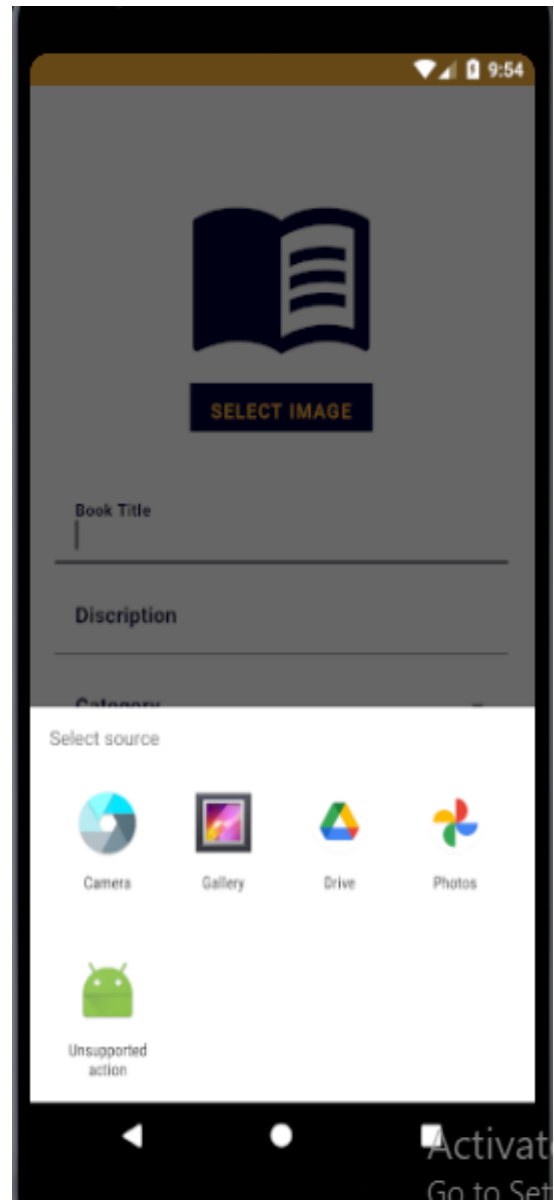
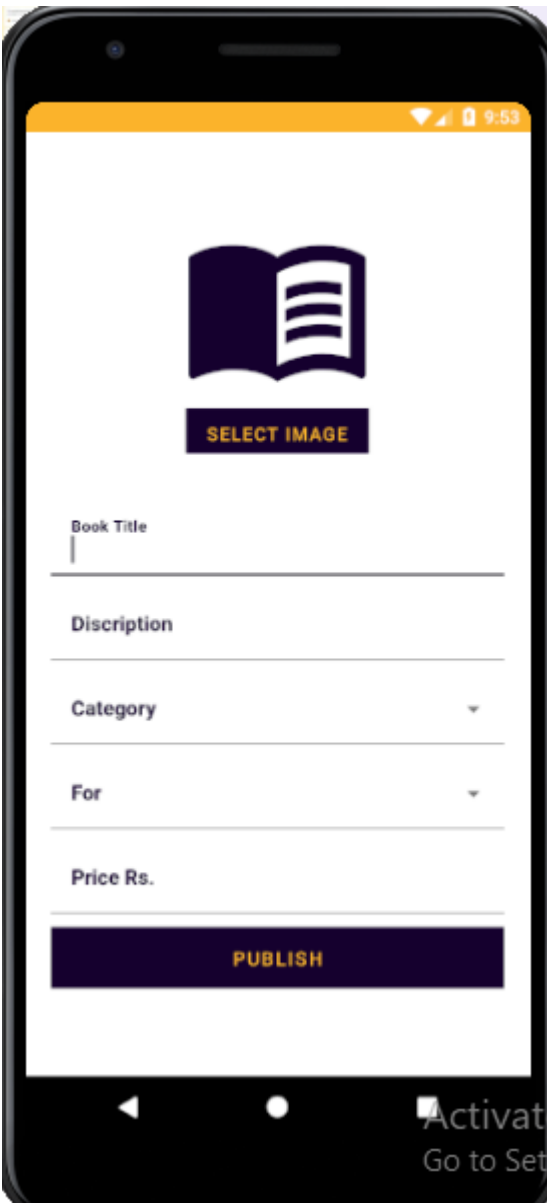


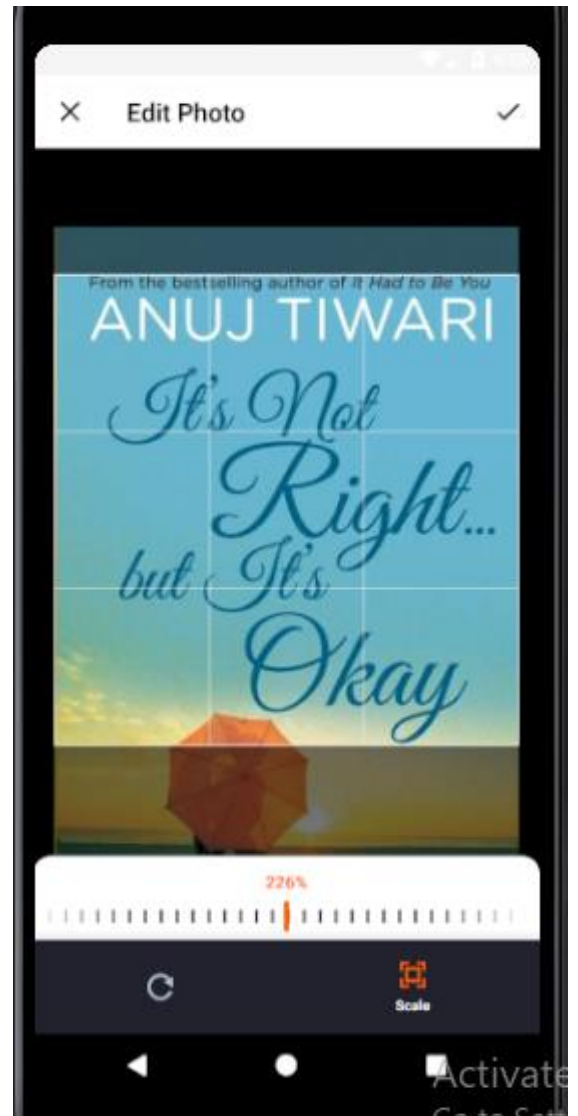
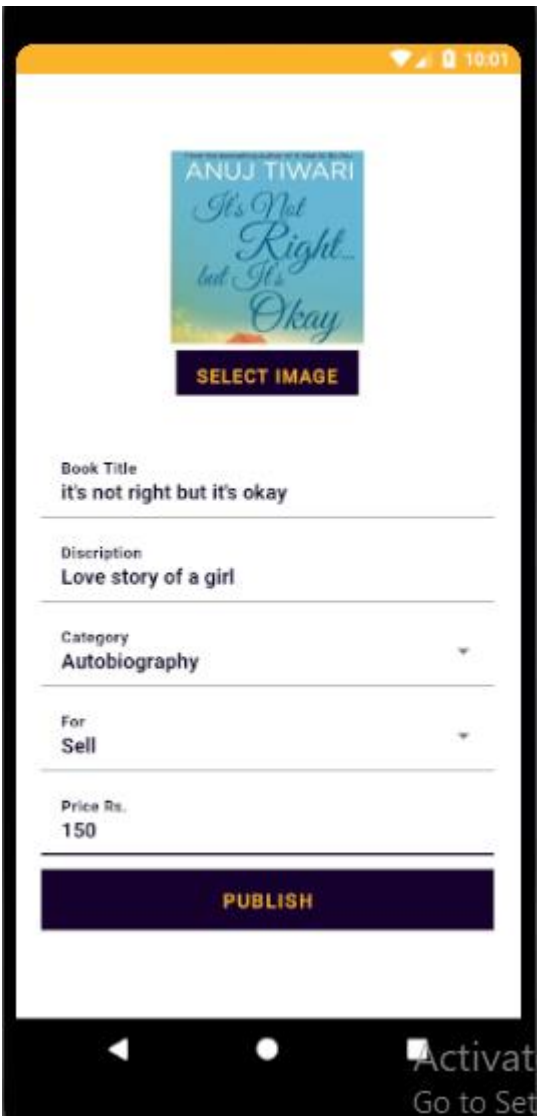
Profile: Profile will show user profile and user can also update details. If data is changed it'll show a toast saying "data has been updated". If data is not changed it'll show a toast saying "data is same and can't be updated".

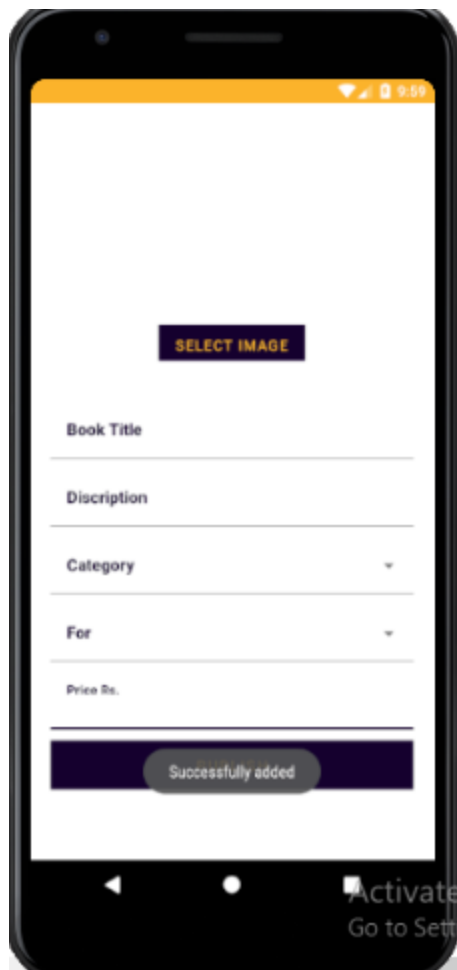




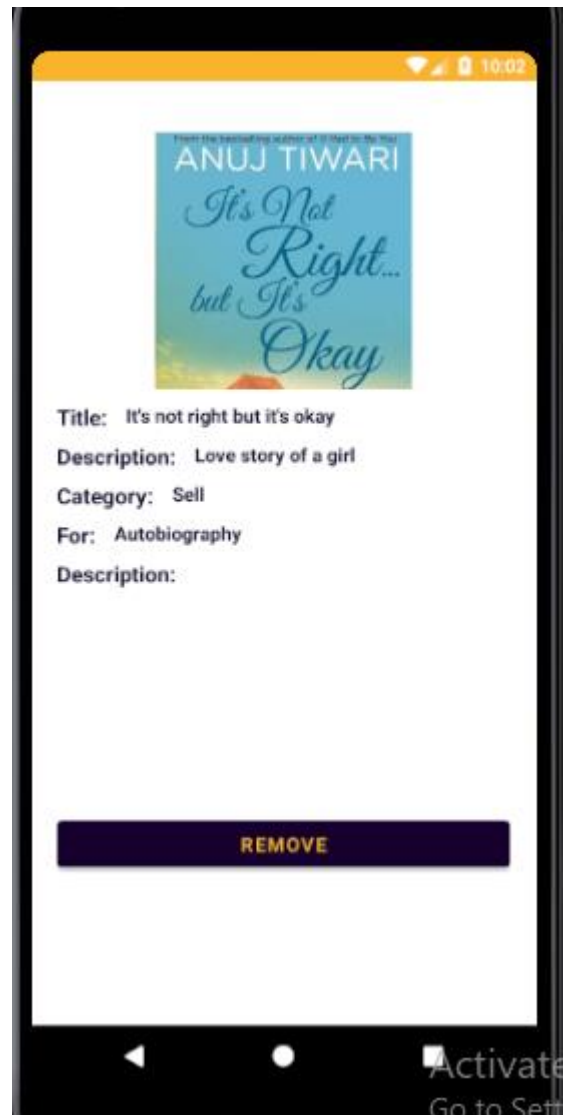
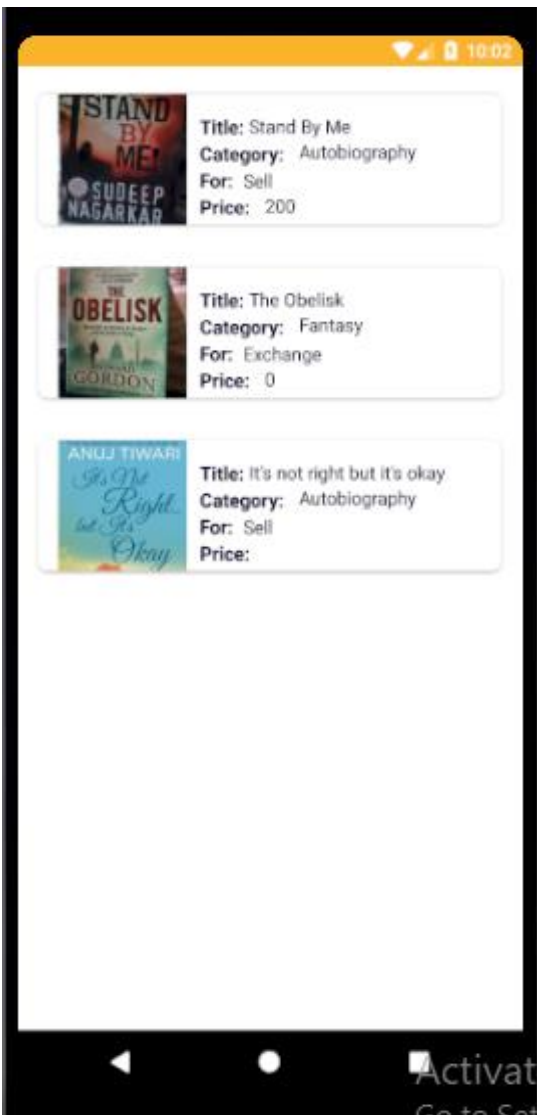
Add Books: User can add books by clicking on the Add book menu. User can also select book from gallery or camera and can crop, rotate images.



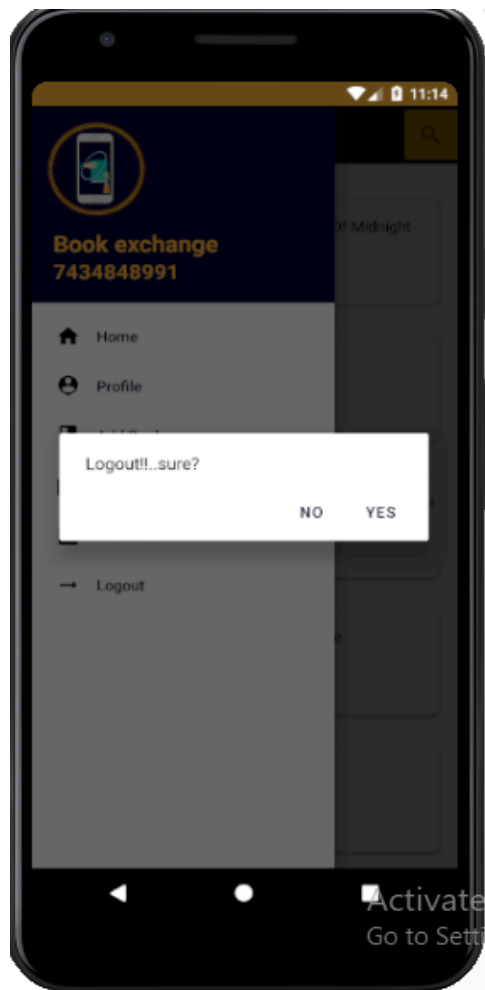




My Books: It'll show added books by user and user can also remove it by clicking on a particular image.



Logout: User will logout by clicking on the logout menu and redirect to Login screen.



Android Manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.book_holic">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Book_holic">
        <activity android:name=".RemoveBook"></activity>
        <activity android:name=".MyBooks" />
        <activity android:name=".contact" />
        <activity android:name=".forgetPassword" />
        <activity android:name=".VerifyPhoneNo" />
        <activity android:name=".sidebarNewUser" />
        <activity
            android:name="com.yalantis.ucrop.UCropActivity"
            android:screenOrientation="portrait"
            android:theme="@style/Theme.AppCompat.Light.NoActionBar" />
        <activity android:name=".recycleview" />
        <activity android:name=".dummy" />
        <activity android:name=".about" />
        <activity android:name=".addBook" />
        <activity android:name=".UserProfile" />
        <activity android:name=".sidebar" />
        <activity android:name=".SignUp" />
        <activity android:name=".Login" />
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <meta-data
            android:name="preloaded_fonts"
            android:resource="@array/preloaded_fonts" />
    </application>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

```

    <uses-permission android:name="android.permission.CALL_PHONE" />

    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

</manifest>

```

Activity about.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:orientation="vertical"
    android:background="@color/one"
    tools:context=".about">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:layout_marginTop="30dp"
        android:src="@drawable/finalbluelogo"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="25sp"
        android:layout_marginTop="50dp"
        android:textColor="@color/three"
        android:textStyle="bold"
        android:justificationMode="inter_word"
        android:padding="20dp"
        android:text="This App focuses on the users who wants to exchange,sell or buy
books at a cheaper rate and bring people closer who likes to read books."/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="20dp"
        android:layout_marginTop="10dp">

        <com.google.android.material.card.MaterialCardView
            android:layout_width="0dp"
            android:layout_height="125dp"
            android:layout_marginTop="30dp"
            android:layout_weight="1"
            app:cardBackgroundColor="@color/white"
            app:cardElevation="10dp"
            >

            <RelativeLayout
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:orientation="vertical">

```

```

        <TextView
            android:id="@+id/name1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="135dp"
            android:layout_marginTop="12dp"
            android:includeFontPadding="false"
            android:text="Admin"
            android:textColor="@color/one"
            android:textSize="30dp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/email1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/name1"
            android:layout_centerHorizontal="true"
            android:includeFontPadding="false"
            android:text="admin123@gmail.com"
            android:textColor="@color/three"
            android:textSize="25dp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/phone1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/email1"
            android:layout_centerHorizontal="true"
            android:includeFontPadding="false"
            android:text="+91 9521684222"
            android:textColor="@color/three"
            android:textSize="25dp"
            android:textStyle="bold" />
    </RelativeLayout>

</com.google.android.material.card.MaterialCardView>

</LinearLayout>

</LinearLayout>

```

Activity add book.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=".UserProfile"
    >

```



```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="270dp"
        android:background="@color/white"
        android:padding="20dp">

        <ImageView
            android:id="@+id/book_img"
            android:layout_width="150dp"
            android:layout_height="150dp"
            android:src="@drawable/book"
            android:layout_centerHorizontal="true"
            android:layout_centerVertical="true"
        />

        <Button
            android:id="@+id/select_image"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/book_img"
            android:layout_centerHorizontal="true"
            android:background="@color/teal_700"
            android:text="Select Image"
            android:layout_marginTop="5dp"
            android:textColor="@color/three" />

    </RelativeLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="20dp"
        android:orientation="vertical">

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Book Title"
            android:id="@+id/bookTitle"
            android:textColorHint="@color/one"
            android:layout_marginBottom="10dp">

            <com.google.android.material.textfield.TextInputEditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:drawablePadding="10dp"

```

```

        android:textColor="@color/one"
        android:textStyle="bold"
        android:background="@color/white"
    />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Discription"
    android:textColorHint="@color/one"
    android:id="@+id/bookDesc"
    android:layout_marginBottom="10dp">

    <com.google.android.material.textfield.TextInputEditText

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawablePadding="10dp"
        android:textColor="@color/one"
        android:textStyle="bold"
        android:background="@color/white"

    />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Category"
    android:textColorHint="@color/one"
    app:boxBackgroundColor="@color/white"
style="@style/Widget.MaterialComponents.TextInputLayout.FilledBox.ExposedDropDownMenu"
    android:layout_marginBottom="10dp">

    <AutoCompleteTextView
        android:id="@+id/category1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:enabled="false"
        android:inputType="none"
        android:textColor="@color/one"
        android:textStyle="bold" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="For"
    android:textColorHint="@color/one"

```

```

        app:boxBackgroundColor="@color/white"
style="@style/Widget.MaterialComponents.TextInputLayout.FilledBox.ExposedDropDownMenu"
"

        android:layout_marginBottom="10dp">

        <AutoCompleteTextView
            android:id="@+id/forSellExchange"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:enabled="false"
            android:inputType="none"
            android:textColor="@color/one"
            android:textStyle="bold"
            />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Price Rs."
        android:id="@+id/amount"
        android:textColorHint="@color/one"
        android:layout_marginBottom="10dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:textColor="@color/one"
            android:textStyle="bold"
            android:background="@color/white"
            />
    </com.google.android.material.textfield.TextInputLayout>

    <Button
        android:id="@+id/publish"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Publish"

        android:textSize="15dp"
        android:background="@color/teal_700"/>

    </LinearLayout>

</LinearLayout>

</ScrollView>

```

Activity_contact.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:padding="20sp"
    android:layout_height="match_parent"
    tools:context=".contact">
    <ImageView
        android:id="@+id/book_image"
        android:layout_centerHorizontal="true"
        android:layout_width="200sp"
        android:background="@drawable/colortool"
        android:layout_height="200sp"
        android:layout_marginTop="20dp"
    />

    <TextView
        android:id="@+id/lblbkTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_image"
        android:text="Title:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="16sp"
        android:textStyle="bold"
    />

    <TextView
        android:id="@+id/book_title"
        android:layout_marginTop="10dp"
        android:layout_toRightOf="@id/lblbkTitle"
        android:layout_marginLeft="15dp"
        android:layout_width="match_parent"
        android:layout_below="@id/book_image"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:textStyle="bold"
        android:text="Title"
        android:textColor="@color/one"/>

    <TextView
        android:id="@+id/lblbkdesc"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_title"
        android:text="Description:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="16sp"
        android:textStyle="bold"
    />
```

```

<TextView
    android:id="@+id/book_description"
    android:layout_marginTop="10dp"
    android:layout_width="match_parent"
    android:layout_toRightOf="@id/lblbkdesc"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:textSize="15sp"
    android:textStyle="bold"
    android:layout_below="@id/book_title"
    android:text="Description"
    android:textColor="@color/one"/>

<TextView
    android:id="@+id/lblbkcat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/book_description"
    android:text="Category:"
    android:textColor="@color/one"
    android:layout_marginTop="10dp"
    android:textSize="16sp"
    android:textStyle="bold"
/>
<TextView
    android:id="@+id/book_category"
    android:layout_marginTop="10dp"
    android:layout_toRightOf="@id/lblbkcat"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"
    android:textSize="15sp"
    android:textStyle="bold"
    android:layout_below="@id/book_description"
    android:text="Category"
    android:textColor="@color/one"/>

<TextView
    android:id="@+id/lblbkfor"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/book_category"
    android:text="For:"
    android:textColor="@color/one"
    android:layout_marginTop="10dp"
    android:textSize="16sp"
    android:textStyle="bold"
/>
<TextView
    android:id="@+id/book_for"
    android:layout_marginTop="10dp"
    android:layout_toRightOf="@id/lblbkfor"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="15dp"

```

```

        android:textSize="15sp"
        android:textStyle="bold"
        android:layout_below="@id/book_category"
        android:text="For"
        android:textColor="@color/one"/>

<TextView
    android:id="@+id/lblbkprice"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/book_for"
    android:text="Price: "
    android:textColor="@color/one"
    android:layout_marginTop="10dp"
    android:textSize="16sp"
    android:textStyle="bold"
/>
<TextView
    android:id="@+id/book_price"
    android:layout_marginTop="10dp"
    android:layout_toRightOf="@id/lblbkprice"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="15sp"
    android:layout_marginLeft="15dp"
    android:textStyle="bold"
    android:layout_below="@id/book_for"
    android:text="Price"
    android:textColor="@color/one"/>

<Button
    android:id="@+id/contact"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="550dp"
    android:layout_centerHorizontal="true"
    android:textSize="15dp"
    android:text="Call"
/>
</RelativeLayout>

```

Activity forgot password.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:layout_margin="50dp"
    tools:context=".VerifyPhoneNo"
    android:orientation="vertical">

    <EditText
        android:id="@+id/newPassword"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter new password"
        android:textColorHint="@color/one"
        android:layout_marginTop="50dp"
        tools:ignore="MissingConstraints"
        android:textColor="@color/one"/>
    <Button
        android:id="@+id/updatePassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:textSize="15dp"
        android:layout_gravity="center"
        android:text="Update"/>
</LinearLayout>

```

Activity_login.xml

```

<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.circularreveal.CircularRevealRelativeLayout
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=".Login">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:padding="20dp">

            <ImageView
                android:id="@+id/imageView"
                android:layout_width="150dp"
                android:layout_height="150dp"
                android:layout_centerHorizontal="true"
                android:transitionName="logo_image"
                app:srcCompat="@drawable/finalwhitelogo"
                android:layout_alignParentTop="true"
                android:paddingTop="5dp"/>

            <TextView
                android:id="@+id/book"
                android:layout_width="match_parent"
                android:layout_height="130dp"
                android:text="Bookoholic"
                android:textSize="60dp"
                android:textColor="@color/one"
                android:transitionName="logo_text"

```

```

        android:fontFamily="cursive"
        android:textAlignment="center"
        android:layout_marginTop="10dp"
        android:layout_below="@id/imageView"
    />

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/username"
        android:hint="Phone Number"
        android:textColorHint="@color/one"
        android:transitionName="username"
        android:layout_marginTop="10dp"
        android:layout_below="@id/book"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"

            android:layout_height="wrap_content"
            android:textColor="@color/one"
            android:textStyle="bold"
        />

    </com.google.android.material.textfield.TextInputLayout>
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        app:passwordToggleEnabled="true"
        android:id="@+id/password"
        android:hint="Password"
        android:transitionName="password"
        android:layout_below="@id/username"
        android:textColorHint="@color/one"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/one"
            android:textStyle="bold"
            android:inputType="textPassword"/>

    </com.google.android.material.textfield.TextInputLayout>

    <Button
        android:id="@+id/forget_password"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:background="#00ffffff"
        android:text="Forget Password?"
        android:layout_below="@id/password"
        android:layout_marginTop="5dp"

```



```

        android:layout_marginLeft="165dp"
        android:textColor="@color/one"
        android:elevation="0dp"/>
    <Button
        android:id="@+id/go"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Go"
        android:layout_below="@id/forget_password"
        android:background="#000"
        android:textColor="@color/three"
        android:transitionName="go"
        android:textSize="15dp"
        android:layout_marginTop="5dp"
        android:layout_marginBottom="5dp"/>
    <Button
        android:id="@+id/signup_screen"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/go"
        android:background="#000"
        android:textColor="@color/three"
        android:transitionName="btn_trans"
        android:textSize="15dp"
        android:layout_marginTop="5dp"
        android:text="New user? Sign Up"
    />
</RelativeLayout>
</ScrollView>

</com.google.android.material.circularreveal.CircularRevealRelativeLayout>

```

Activity main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.circularreveal.CircularRevealRelativeLayout
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:background="@color/one"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/logo"
        android:layout_width="300dp"
        android:layout_height="300dp"
        android:src="@drawable/finalbluelogo"
        android:layout_centerHorizontal="true"
        android:layout_marginLeft="10dp"
        android:transitionName="logo_image"
        android:layout_marginRight="10dp"
    />

```

```

        android:layout_marginTop="30dp"
    />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:text="Bookoholic"
        android:textSize="60dp"
        android:textColor="@color/three"
        android:transitionName="logo_text"
        android:fontFamily="cursive"
        android:textAlignment="center"
        android:layout_marginTop="100dp"
        android:layout_below="@id/logo"
    />

</com.google.android.material.circularreveal.CircularRevealRelativeLayout>

```

Activity_my_books.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=".MyBooks">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_marginTop="5dp"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />
</RelativeLayout>

```

Activity_recycleview.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=".recycleview">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"

```

```

        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

Activity remove books.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:padding="20sp"
    android:layout_height="match_parent"
    tools:context=".RemoveBook">
    <ImageView
        android:id="@+id/book_image"
        android:layout_centerHorizontal="true"
        android:layout_width="200sp"
        android:layout_height="200sp"
        android:background="@drawable/colortool"
        android:layout_marginTop="20dp"
        />

    <TextView
        android:id="@+id/lb1bkTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_image"
        android:text="Title:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
        android:textStyle="bold"
        />
    <TextView
        android:id="@+id/book_title"
        android:layout_marginTop="10dp"
        android:layout_toRightOf="@id/lb1bkTitle"
        android:layout_marginLeft="15dp"
        android:layout_width="match_parent"
        android:layout_below="@id/book_image"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:textStyle="bold"
        android:text="Title"
        android:textColor="@color/one"/>

    <TextView
        android:id="@+id/lb1bkdesc"
        android:layout_width="wrap_content"

```

```
        android:layout_height="wrap_content"
        android:layout_below="@id/book_title"
        android:text="Description:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
        android:textStyle="bold"
    />
```

<TextView

```
        android:id="@+id/book_description"
        android:layout_marginTop="10dp"
        android:layout_width="match_parent"
        android:layout_toRightOf="@id/lblbkdesc"
        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
        android:textSize="15sp"
        android:textStyle="bold"
        android:layout_below="@id/book_title"
        android:text="Description"
        android:textColor="@color/one" />
```

<TextView

```
        android:id="@+id/lblbkcat"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_description"
        android:text="Category:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
        android:textStyle="bold"
    />
```

<TextView

```
        android:id="@+id/book_category"
        android:layout_marginTop="10dp"
        android:layout_toRightOf="@id/lblbkcat"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
        android:textSize="15sp"
        android:textStyle="bold"
        android:layout_below="@id/book_description"
        android:text="Category"
        android:textColor="@color/one" />
```

<TextView

```
        android:id="@+id/lblbkfor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_category"
        android:text="For:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
```

```

        android:textStyle="bold"
    />
    <TextView
        android:id="@+id/book_for"
        android:layout_marginTop="10dp"
        android:layout_toRightOf="@id/lblbkfor"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
        android:textSize="15sp"
        android:textStyle="bold"
        android:layout_below="@id/book_category"
        android:text="For"
        android:textColor="@color/one"/>

    <TextView
        android:id="@+id/lblbkprice"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/book_for"
        android:text="Description:"
        android:textColor="@color/one"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
        android:textStyle="bold"
    />
    <TextView
        android:id="@+id/book_price"
        android:layout_marginTop="10dp"
        android:layout_toRightOf="@id/lblbkprice"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:layout_marginLeft="15dp"
        android:textStyle="bold"
        android:layout_below="@id/book_for"
        android:text="Price"
        android:textColor="@color/one"/>

    <Button
        android:id="@+id/remove"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="550dp"
        android:layout_centerHorizontal="true"
        android:textSize="15dp"
        android:text="Remove"
    />
</RelativeLayout>

```

Activity sidebar.xml

```

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

```

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/drawer"
        tools:openDrawer="start"

        xmlns:app="http://schemas.android.com/apk/res-auto">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent">
            <androidx.appcompat.widget.Toolbar
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/toolbar"
                app:title="Bookoholic"

                app:titleTextColor="@color/three"
                android:background="@color/teal_700"
            />
            <androidx.recyclerview.widget.RecyclerView
                android:id="@+id/recyclerView"
                android:layout_below="@id/toolbar"
                android:layout_marginTop="5dp"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                app:layout_constraintBottom_toBottomOf="parent"
                app:layout_constraintEnd_toEndOf="parent"
                app:layout_constraintHorizontal_bias="1.0"
                app:layout_constraintStart_toStartOf="parent"
                app:layout_constraintTop_toTopOf="parent"
                app:layout_constraintVertical_bias="0.0" />
            </RelativeLayout>

            <com.google.android.material.navigation.NavigationView
                android:layout_width="wrap_content"
                android:layout_height="match_parent"
                android:id="@+id/navmenu"
                app:menu="@menu/iconmenu"
                app:itemTextColor="@color/teal_700"
                app:headerLayout="@layout/profile_header"
                android:layout_gravity="start"/>
        </androidx.drawerlayout.widget.DrawerLayout>

```

Activity sidebar new user.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer1"
    tools:openDrawer="start"
    xmlns:app="http://schemas.android.com/apk/res-auto">

```

```

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <androidx.appcompat.widget.Toolbar
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/toolbar1"
        app:title="Bookoholic"
        app:titleTextColor="@color/three"
        android:background="@color/teal_700"
        />
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_below="@id/toolbar1"
        android:layout_marginTop="5dp"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />
</RelativeLayout>

<com.google.android.material.navigation.NavigationView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:id="@+id/navmenu1"
    app:menu="@menu/iconmenu1"
    app:itemTextColor="@color/one"
    app:headerLayout="@layout/profile_header1"
    android:layout_gravity="start"/>
</androidx.drawerlayout.widget.DrawerLayout>

```

Activity sign_up.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:orientation="vertical"
    android:padding="20dp"
    tools:context=".SignUp">

    <ImageView
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:layout_gravity="center"
        android:transitionName="logo_image"

```

```

        android:src="@drawable/finalwhitelogo"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="welcome"
    android:transitionName="logo_text"
    android:textColor="@color/one"
    android:textSize="30sp"
    android:textStyle="bold"
    android:layout_marginTop="-5dp"
/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:transitionName="logo_text"
    android:textColor="@color/three"
    android:textStyle="bold"
    android:text="sign up to start new journey"
    android:textSize="18sp"/>
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_marginTop="20dp">
    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:hint="Full Name"
        android:textColorHint="@color/one"
        android:id="@+id/name"
    >
        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="text"
            android:textColor="@color/one"

        />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:hint="Username"
        android:textColorHint="@color/one"
        android:transitionName="username"
        android:id="@+id/username"
        app:counterMaxLength="15">

```



```

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="text"

        />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColorHint="@color/one"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:hint="Email"
        android:id="@+id/email"
    >

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textEmailAddress"
        />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:hint="Phone number"
        android:textColorHint="@color/one"
        android:id="@+id/phone"
    >

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="number"
        />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:hint="Password"
        android:textColorHint="@color/one"
        android:transitionName="password"
        android:layout_marginBottom="20dp"
        app:passwordToggleEnabled="true"
        android:id="@+id/password"
    >

        <com.google.android.material.textfield.TextInputEditText

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
    />

</com.google.android.material.textfield.TextInputLayout>

<Button
    android:id="@+id/go"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000"
    android:layout_gravity="center"
    android:text="Go"
/>

<Button
    android:id="@+id/goToLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:background="#000"
    android:text="Already have an account? log in"
    android:transitionName="btn_trans" />
</LinearLayout>
</ScrollView>
</LinearLayout>

```

Activity use 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=".UserProfile"
    >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="250dp"
            android:padding="20dp"
            android:background="@color/teal_700">

            <ImageView
                android:id="@+id/profile_image"
                android:layout_width="150dp"
                android:layout_height="150dp"
                android:layout_centerHorizontal="true"
                android:src="@drawable/finalbluelogo"
            />
        </RelativeLayout>
    </LinearLayout>
</ScrollView>

```

```

        android:layout_centerVertical="true"/>

</RelativeLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="20dp"
    android:orientation="vertical">

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Full name"
        android:textColor="@color/one"
        android:id="@+id/full_name"
        android:textColorHint="@color/one"
        android:layout_marginBottom="10dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:background="@color/white"
            android:drawableLeft="@drawable/user_profile"
            android:textStyle="bold"
            android:textColor="@color/one"
            android:text=""
            />

    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="E-mail"
        android:textColor="@color/one"
        android:id="@+id/email"
        android:textColorHint="@color/one"
        android:layout_marginBottom="10dp">

        <com.google.android.material.textfield.TextInputEditText
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:background="@color/white"
            android:drawableLeft="@drawable/email"
            android:textStyle="bold"
            android:textColor="@color/one"
            android:text=""
            />
    </com.google.android.material.textfield.TextInputLayout>
</LinearLayout>

```

```

        </com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Password"
            android:textColor="@color/one"
            android:id="@+id/password"
            android:textColorHint="@color/one"
            android:layout_marginBottom="10dp">

            <com.google.android.material.textfield.TextInputEditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:drawablePadding="10dp"
                android:background="@color/white"
                android:drawableLeft="@drawable/password"
                android:textStyle="bold"
                android:textColor="@color/one"
                android:text=""
            />

        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Update"
            android:id="@+id/update"
            android:textSize="15dp"
            android:background="@color/teal_700"/>

    </LinearLayout>
</LinearLayout>
</ScrollView>

```

Activity verify phone no.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:layout_margin="50dp"

    tools:context=".VerifyPhoneNo"
    android:orientation="vertical">

    <EditText
        android:id="@+id/verification_code_entered_by_user"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"

```

```

        android:hint="Enter otp"
        android:textColorHint="@color/one"
        android:textColor="@color/one"
        tools:ignore="MissingConstraints" />

        <Button
            android:id="@+id/verify_btn"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:textSize="15dp"
            android:layout_gravity="center"
            android:text="verify"/>

        <ProgressBar
            android:id="@+id/progress_bar"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="20dp"/>
    </LinearLayout>

```

Forcategory.xml

```

<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ellipsize="end"
    android:padding="16dp"
    android:maxLines="1"
    android:background="@color/white"
    android:textAppearance="?attr/textAppearanceButton"
    >

</TextView>

```

Profile_header.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.LinearLayoutCompat
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/teal_700"
    android:orientation="vertical"
    >
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="180dp"
        android:layout_marginTop="10sp"
        android:layout_marginLeft="10sp"
        android:background="@color/teal_700">

        <ImageView
            android:id="@+id/profile_image"

```

```

        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_marginBottom="10sp"
        android:src="@drawable/finalbluelogo"
    />

    <TextView
        android:id="@+id/full_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_marginLeft="10dp"
        android:includeFontPadding="false"
        android:layout_below="@id/profile_image"
        android:textSize="22dp"
        android:textStyle="bold"
        android:textColor="@color/three"
        android:text="Book exchange"

    />

    <TextView
        android:id="@+id/about"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:includeFontPadding="false"
        android:layout_marginTop="2sp"
        android:textColor="@color/three"
        android:textStyle="bold"
        android:layout_below="@+id/full_name"
        android:text="user123"
        android:textSize="20dp"
    />
</RelativeLayout>
</androidx.appcompat.widget.LinearLayoutCompat>

```

Profile header1.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.LinearLayoutCompat
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/teal_700"
    android:orientation="vertical"
    >
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="180dp"
        android:padding="20dp"
        android:background="@color/teal_700">

        <ImageView

```

```

        android:id="@+id/profile_image"
        android:layout_width="120dp"
        android:layout_height="130dp"
        android:src="@drawable/finalbluelogo"
    />

</RelativeLayout>

</androidx.appcompat.widget.LinearLayoutCompat>

```

Recycler item.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardElevation="2dp"
    app:cardCornerRadius="8dp"
    android:id="@+id/cardView"
    android:layout_margin="16dp">

    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="16dp"
    >

        <ImageView
            android:id="@+id/bookLogo"
            android:layout_width="100dp"
            android:layout_centerVertical="true"
            android:layout_height="100dp"
            android:scaleType="centerCrop"
            android:src="@drawable/finalwhitelogo"/>

        <TextView
            android:id="@+id/lblTitle"
            android:layout_width="wrap_content"
            android:layout_toRightOf="@+id/bookLogo"
            android:layout_marginLeft="10sp"
            android:layout_height="wrap_content"
            android:text="Title: "
            android:textColor="@color/one"
            android:layout_marginTop="15dp"
            android:textSize="15sp"
            android:textStyle="bold"/>

        <TextView
            android:id="@+id/title1"
            android:layout_toRightOf="@+id/lblTitle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Title"
            android:textColor="@color/one"

```

```
        android:layout_marginTop="15dp"
        android:textSize="14sp" />
```

```
<TextView
```

```
    android:id="@+id/lblDesc"
    android:layout_below="@id/lblTitle"
    android:layout_toRightOf="@+id/bookLogo"
    android:layout_marginLeft="10sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Description: "
    android:textColor="@color/one"
    android:textSize="15sp"
    android:textStyle="bold"/>
```

```
<TextView
```

```
    android:id="@+id/desc1"
    android:layout_below="@+id/title1"
    android:layout_toRightOf="@id/lblDesc"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Description "
    android:textColor="@color/one"
    android:textSize="14sp" />
```

```
<TextView
```

```
    android:id="@+id/lblCategory"
    android:layout_below="@+id/lblDesc"
    android:layout_toRightOf="@+id/bookLogo"
    android:layout_marginLeft="10sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Category: "
    android:textColor="@color/one"
    android:textSize="15sp"
    android:textStyle="bold"/>
```

```
<TextView
```

```
    android:id="@+id/category1"
    android:layout_below="@+id/desc1"
    android:layout_toRightOf="@+id/lblCategory"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="category "
    android:layout_marginLeft="8dp"
    android:layout_marginTop="0dp"
    android:textColor="@color/one"
    android:textSize="14sp" />
```

```
<TextView
```

```
    android:id="@+id/lblFor"
    android:layout_below="@id/lblCategory"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```



```
        android:layout_toRightOf="@+id/bookLogo"
        android:layout_marginLeft="10sp"
        android:text="For:"
        android:textColor="@color/one"
        android:textSize="15sp"
        android:textStyle="bold"/>
```

<TextView

```
        android:id="@+id/for1"
        android:layout_below="@id/category1"
        android:layout_toRightOf="@+id/lblFor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="For"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="3dp"
        android:textColor="@color/one"
        android:textSize="14sp" />
```

<TextView

```
        android:id="@+id/lblPrice"
        android:layout_below="@id/lblFor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@+id/bookLogo"
        android:layout_marginLeft="10sp"
        android:text="Price: "
        android:textColor="@color/one"
        android:textSize="15sp"
        android:textStyle="bold"/>
```

<TextView

```
        android:id="@+id/price1"
        android:layout_below="@id/for1"
        android:layout_toRightOf="@id/lblPrice"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="20"
        android:layout_marginLeft="8dp"
        android:textColor="@color/one"
        android:textSize="14sp" />
```

<TextView

```
        android:id="@+id/lblPhone"
        android:layout_below="@id/lblPrice"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@+id/bookLogo"
        android:layout_marginLeft="10sp"
        android:text="Phone: "
        android:textColor="@color/one"
        android:textSize="15sp"
        android:textStyle="bold"/>
```

<TextView

```

        android:id="@+id/Phone1"
        android:layout_below="@id/price1"
        android:layout_toRightOf="@id/lblPhone"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="123456"
        android:layout_marginLeft="8dp"
        android:textColor="@color/one"
        android:textSize="14sp" />

    </RelativeLayout>

</androidx.cardview.widget.CardView>

```

Sellexchange.xml

```

<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ellipsize="end"
    android:padding="16dp"
    android:maxLines="1"
    android:background="@color/white"
    android:textAppearance="?attr/textAppearanceSubtitle1"
    >

</TextView>

```

Menu>iconmenu.xml

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_home"
        android:icon="@drawable/home"
        android:title="Home"/>

    <item android:id="@+id/profile"
        android:icon="@drawable/ic_baseline_person_pin_24"
        android:title="Profile"/>

    <item android:id="@+id/add_book"
        android:icon="@drawable/ic_baseline_book_24"
        android:title="Add Book"/>

    <item android:id="@+id/my_books"
        android:icon="@drawable/ic_baseline_library_books_24"
        android:title="My Books"/>

    <item android:id="@+id/about"
        android:icon="@drawable/ic_baseline_help_center_24"
        android:title="About Us"/>

    <item android:id="@+id/logout"

```

```
        android:icon="@drawable/ic_baseline_arrow_right_alt_24"
        android:title="Logout"/>
```

```
</menu>
```

Menu>iconmenu1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_home1"
        android:icon="@drawable/ic_baseline_home_24"
        android:title="Home"/>

    <item android:id="@+id/about1"
        android:icon="@drawable/ic_baseline_help_center_24"
        android:title="About Us"/>

    <item android:id="@+id/login"
        android:icon="@drawable/ic_baseline_arrow_right_alt_24"
        android:title="Login"/>
</menu>
```

Menu>searchmenu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:tools="http://schemas.android.com/tools"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item android:id="@+id/search"
        android:title="search data"
        android:icon="@drawable/search2"
        app:showAsAction="always"
        app:actionViewClass="android.widget.SearchView" >
    </item>
</menu>
```

Values>color.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#130B30</color>
    <color name="one">#130B30</color>
    <color name="two">#20D0C3</color>
    <color name="three">#FBB329</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>
```

Values>string.xml

```
<resources>
    <string name="app_name">Book_holic</string>
    <string name="open">open</string>
    <string name="close">close</string>
</resources>
```

Themes>themes.xml

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.Book_holic"
parent="Theme.MaterialComponents.DayNight.NoActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/one</item>
        <item name="colorPrimaryVariant">@color/two</item>
        <item name="colorOnPrimary">@color/three</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/two</item>
        <item name="colorOnSecondary">@color/three</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor" tools:targetApi="1">@color/three</item>
        <!-- Customize your theme here. -->
        <item name="android:windowContentTransitions">true</item>
        <item name="android:actionOverflowButtonStyle">@style/myTheme</item>
    </style>

    <style name="ActionBarTheme"
parent="@style/Widget.AppCompat.Light.ActionBar.Solid.Inverse">
        <item name="android:background">@color/three</item>
        <item name="background">@color/two</item>
    </style>
    <style name="buttontheme" >
        <item name="android:background">@color/three</item>
        <item name="android:textSize">19dp</item>
        <item name="android:layout_margin">8dp</item>
        <item name="android:padding">8dp</item>
        <item name="android:textColor">@color/two</item>
    </style>
    <style name="ToggleTheme"
parent="@style/Widget.AppCompat.Light.ActionBar.Solid.Inverse">
        <item name="android:background">@color/three</item>
        <item name="background">@color/two</item>
    </style>

    <style name="myTheme" parent="android:style/Widget.Holo.ActionButton.Overflow">
        <item name="android:src">@drawable/search2</item>
    </style>
</resources>
```

Themes>themes.xml (dark)

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.Book_holic"
parent="Theme.MaterialComponents.DayNight.NoActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/purple_200</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/black</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_200</item>
        <item name="colorOnSecondary">@color/black</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor"
tools:targetApi="1">?attr/colorPrimaryVariant</item>
        <!-- Customize your theme here. -->
        <item name="android:windowContentTransitions">true</item>
    </style>
</resources>
```

About.java

```
package com.example.book_holic;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class about extends AppCompatActivity {

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

Addbook.java

```
package com.example.book_holic;

import android.content.Context;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AutoCompleteTextView;
import android.widget.Button;
import android.widget.ImageView;
```

```

import android.widget.Toast;

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

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.textfield.TextInputLayout;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;
import com.myhexaville.smartimagepicker.ImagePicker;
import com.myhexaville.smartimagepicker.OnImagePickedListener;
import com.squareup.picasso.Picasso;
import com.yalantis.ucrop.UCrop;

import java.io.File;
import java.io.IOException;
import java.util.HashMap;

public class addBook extends AppCompatActivity {
    AutoCompleteTextView autoCompleteTextView1,autoCompleteTextView2;
    private Button selectImage,publish;
    private ImageView imageView;
    String phone;
    Uri resultUri;
    ImagePicker imagePicker;
    DatabaseReference root;
    TextInputLayout title,desc,amt;
    private final int PERMISSION_ALL = 1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_add_book);
        //
        title=findViewById(R.id.bookTitle);
        desc=findViewById(R.id.bookDesc);
        amt=findViewById(R.id.amount);
        autoCompleteTextView1 = findViewById(R.id.forSellExchange);
        autoCompleteTextView2 = findViewById(R.id.category1);
        //
        selectImage = findViewById(R.id.select_image);
        publish = findViewById(R.id.publish);
        imageView = findViewById(R.id.book_img);
        final String[] PERMISSIONS = {
            android.Manifest.permission.WRITE_EXTERNAL_STORAGE,
            android.Manifest.permission.READ_EXTERNAL_STORAGE,
            android.Manifest.permission.CAMERA
        };

        phone = getIntent().getStringExtra("phone");
        //System.out.println(phone);
    }

```

```

        root= FirebaseDatabase.getInstance().getReference("book").child(phone);

        String []option = {"Exchange","Sell"};
        ArrayAdapter arrayAdapter = new
ArrayAdapter(this,R.layout.sellexchange,option);
        // autoCompleteTextView1.setText(arrayAdapter.getItem(0).toString(),false);
        autoCompleteTextView1.setAdapter(arrayAdapter);

        String []op = {"Action &
Adventure","Children's","Crime","Drama","Fantasy","Horror","Mystery","Sci-
Fi","Thriller","Autobiography",
"Business/Economics","Cookbook","Dictionary","Health/Fitness","Philosophy","Travel"};
        ArrayAdapter arrAdapter = new ArrayAdapter(this,R.layout.forcategory,op);
        // autoCompleteTextView2.setText(arrAdapter.getItem(0).toString(),false);
        autoCompleteTextView2.setAdapter(arrAdapter);

        publish.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                upload(resultUri);
            }
        });

        selectImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (!hasPermissions(addBook.this, PERMISSIONS)) {
                    ActivityCompat.requestPermissions(addBook.this, PERMISSIONS,
PERMISSION_ALL);
                }
                else
                {
                    imagePicker.choosePicture(true /*show camera intents*/);
                }
            }
        });
        imagePicker = new ImagePicker(this, /* activity non null*/
            null, /* fragment nullable*/
            new OnImagePickedListener() {
                @Override
                public void onImagePicked(Uri imageUri) {
                    UCrop.of(imageUri, getTempUri())
                        .withAspectRatio(1, 1)
                        .start(addBook.this);
                }
            }
        });

    }
    private Uri getTempUri()
    {
        String dir = Environment.getExternalStorageDirectory()+File.separator+"Temp";
        File dirFile = new File(dir);
        dirFile.mkdir();
    }

```

```

        String file = dir+File.separator+"temp.jpg";
        File tempFile = new File(file);
        try {
            tempFile.createNewFile();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return Uri.fromFile(tempFile);
    }

    public static boolean hasPermissions(Context context, String... permissions) {
        if (context != null && permissions != null) {
            for (String permission : permissions) {
                if (ActivityCompat.checkSelfPermission(context, permission) !=
PackageManager.PERMISSION_GRANTED) {
                    return false;
                }
            }
        }
        return true;
    }

    @Override
    public void onRequestPermissionsResult(int requestCode, String permissionsList[],
int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissionsList, grantResults);
        imagePicker.handlePermission(requestCode, grantResults);
        switch (requestCode) {
            case PERMISSION_ALL:{
                if (grantResults.length > 0) {
                    boolean flag=true;
                    for (int i=0;i<permissionsList.length;i++) {
                        if(grantResults[i] == PackageManager.PERMISSION_DENIED){
                            flag = false;
                        }
                    }
                    if(flag)
                    {
                        imagePicker.choosePicture(true /*show camera
intents*/);// imagePicker.choosePicture(true);
                    }
                }
            }
            return;
        }
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        imagePicker.handleActivityResult(resultCode,requestCode, data);
        if (resultCode == RESULT_OK && requestCode == UCrop.REQUEST_CROP) {
            resultUri = UCrop.getOutput(data);
        }
    }

```



```

        imageView.setImageURI(null);
        imageView.setImageURI(resultUri);
    } else if (resultCode == UCrop.RESULT_ERROR) {
        final Throwable cropError = UCrop.getError(data);
    }
}

void upload(Uri uri) {
    if(!validateTitle())
    { return; }
    else if( !validateDesc()){ return;}
    else if(!validateAmt()) {return;}
    else
    {
        //loader.setVisibility(View.VISIBLE);
        String strTitle, strDesc, strCategory, strType, strAmt;
        strTitle = title.getEditText().getText().toString();
        strDesc = desc.getEditText().getText().toString();
        strAmt = amt.getEditText().getText().toString();
        strCategory = autoCompleteTextView1.getText().toString();
        strType = autoCompleteTextView2.getText().toString();
        StorageReference riversRef =
        FirebaseStorage.getInstance().getReference().child("Temp/" +
        System.currentTimeMillis() + ".png");

        riversRef.putFile(uri).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception exception) {
                Log.i("error", "onfailure" + exception.getMessage());
            }
        }).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

                riversRef.getDownloadUrl().addOnSuccessListener(new
                OnSuccessListener<Uri>() {
                    @Override
                    public void onSuccess(Uri uri) {

                        final Uri downloadUrl = uri;
                        String url = downloadUrl.toString();
                        //String modelId = root.push().getKey();
                        HashMap<String, String> userMap = new HashMap<>();
                        userMap.put("Title", strTitle);
                        userMap.put("Description", strDesc);
                        userMap.put("Category", strCategory);
                        userMap.put("Type", strType);
                        userMap.put("Amount", strAmt);
                        userMap.put("phone", phone);
                        userMap.put("ImageUrl", url);

                        root.push().setValue(userMap);
                        Picasso.get().load(uri).into(imageView);
                    }
                });
            }
        });
    }
}

```

```

        Toast.makeText(getApplicationContext(), "Successfully added",
Toast.LENGTH_LONG).show();
        title.getEditText().setText("");
        desc.getEditText().setText("");
        amt.getEditText().setText("");
        autoCompleteTextView1.setText("");
        autoCompleteTextView2.setText("");
        imageView.setImageResource(0);
        //imageView.setImageResource(android.R.color.transparent);
    }
});
}
}

private Boolean validateTitle()
{
    String val = title.getEditText().getText().toString();
    if (val.isEmpty())
    {
        title.setError("Field can not be empty");
        return false;
    }
    else
    {
        title.setError(null);
        title.setErrorEnabled(false);
        return true;
    }
}

private Boolean validateDesc()
{
    String val = title.getEditText().getText().toString();
    if (val.isEmpty())
    {
        desc.setError("Field can not be empty");
        return false;
    }
    else
    {
        desc.setError(null);
        desc.setErrorEnabled(false);
        return true;
    }
}

private Boolean validateAmt()
{
    String val = title.getEditText().getText().toString();
    if (val.isEmpty())
    {
        amt.setError("Field can not be empty");
        return false;
    }
    else
    {
        amt.setError(null);

```

```

        amt.setErrorEnabled(false);
        return true;
    }
}

```

Bookdata.java

```

package com.example.book_holic;

public class bookdata {
    String Title,Description,Category,Amount,Type,ImageUrl,phone;

    public bookdata() {
    }

    public bookdata(String title, String description, String category, String amount,
String type, String imageUrl, String phoneNo) {
        Title = title;
        Description = description;
        Category = category;
        Amount = amount;
        Type = type;
        ImageUrl = imageUrl;
        phone = phoneNo;
    }

    public String getTitle() {
        return Title;
    }

    public void setTitle(String title) {
        Title = title;
    }

    public String getDescription() {
        return Description;
    }

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

    public String getCategory() {
        return Category;
    }

    public void setCategory(String category) {
        Category = category;
    }

    public String getAmount() {
        return Amount;
    }
}

```

```

    public void setAmount(String amount) {
        Amount = amount;
    }

    public String getType() {
        return Type;
    }

    public void setType(String type) {
        Type = type;
    }

    public String getImageUrl() {
        return imageUrl;
    }

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

    public String getphone() {
        return phone;
    }

    public void setPhoneNo(String phoneNo) {
        phone = phoneNo;
    }
}

```

Contact.java

```

package com.example.book_holic;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;

public class contact extends AppCompatActivity {
    String imageUrl,booktitle,bookDesc,bookCategory,bookFor,bookPrice,phone;
    ImageView _bookImage;
    Button _contact;
    TextView _bookTitle,_bookDesc,_bookCategory,_bookFor,_bookPrice;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_contact);
    }
}

```

```

//Integer imageUrl = getIntent().getIntExtra("imageUrl",0);
imageUrl = getIntent().getStringExtra("imageUrl");
booktitle = getIntent().getStringExtra("title");
bookDesc = getIntent().getStringExtra("description");
bookCategory = getIntent().getStringExtra("category");
bookFor = getIntent().getStringExtra("for");
bookPrice = getIntent().getStringExtra("price");
phone = getIntent().getStringExtra("phone");

_bookImage = findViewById(R.id.book_image);
_bookTitle = findViewById(R.id.book_title);
_bookDesc = findViewById(R.id.book_description);
_bookCategory = findViewById(R.id.book_category);
_bookFor = findViewById(R.id.book_for);
_bookPrice = findViewById(R.id.book_price);
_contact = findViewById(R.id.contact);

// Glide.with(context).load(book.getImageUrl()).into(holder.imageView);
Glide.with(this).asBitmap().load(imageUrl).into(_bookImage);

_bookTitle.setText(booktitle);
_bookDesc.setText(bookDesc);
_bookCategory.setText(bookFor);
_bookFor.setText(bookCategory);
_bookPrice.setText(bookPrice);

_contact.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_DIAL);
        intent.setData(Uri.parse("tel:+"+"91"+phone));
        startActivity(intent);
    }
});
}
}

```

Forgetpassword.java

```

package com.example.book_holic;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

```

```

public class forgetPassword extends AppCompatActivity {
    EditText newPassword;
    Button updatepassword;

    String phoneNo;
    DatabaseReference reference;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forget_password);
        reference = FirebaseDatabase.getInstance().getReference("users");
        newPassword = findViewById(R.id.newPassword);
        updatepassword = findViewById(R.id.updatePassword);

        phoneNo = getIntent().getStringExtra("phoneNo");
        //_password = getIntent().getStringExtra("password");

        updatepassword.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                update(v);
            }
        });
    }
    public void update(View view){
reference.child(phoneNo).child("password").setValue(newPassword.getText().toString())
;
        Intent intent = new Intent(forgetPassword.this, Login.class);
        startActivity(intent);
        finish();
    }
}

```

Login.java

```

package com.example.book_holic;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.annotation.SuppressLint;
import android.app.ActivityOptions;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.media.Image;
import android.os.Bundle;
import android.util.Pair;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import com.google.android.material.textfield.TextInputLayout;

```

```

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

public class Login extends AppCompatActivity {
    Button callSignUp, btnlogin, forgetPassword;
    ImageView image;
    TextView logoText;
    TextInputLayout username, password;
    int flag = 1;

    @SuppressWarnings("ResourceType")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.activity_login);
        logoText = findViewById(R.id.book);
        image = findViewById(R.id.imageView);
        username = findViewById(R.id.username);
        password = findViewById(R.id.password);
        btnlogin = findViewById(R.id.go);
        forgetPassword = findViewById(R.id.forget_password);
        callSignUp = findViewById(R.id.signup_screen);
        forgetPassword.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String val = username.getText().toString();
                String pass = password.getText().toString();
                if (val.isEmpty())
                {
                    username.setError("please enter username");
                    username.requestFocus();
                    return;
                }
                else {
                    Intent intent = new Intent(getApplicationContext(),
VerifyPhoneNo.class);
                    intent.putExtra("flag", flag);
                    intent.putExtra("phoneNo", val);
                    //intent.putExtra("password", pass);
                    startActivity(intent);
                }
            }
        });
        callSignUp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

```

```

        Intent intent= new Intent(Login.this,SignUp.class);
        Pair[] pairs = new Pair[5];

        pairs[0] = new Pair<View,String>(image,"logo_name");
        pairs[1] = new Pair<View,String>(logoText,"logo_text");
        pairs[2] = new Pair<View,String>(username,"username");
        pairs[3] = new Pair<View,String>(password,"password");
        pairs[4] = new Pair<View,String>(btnlogin,"btn_trans");
        ActivityOptions options =
        ActivityOptions.makeSceneTransitionAnimation(Login.this,pairs);

        startActivity(intent,options.toBundle());
        // finish();
    }
});
btnlogin.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(!validateUsername())
        {
            return;
        }
        else if(!validatePassword())
        {
            return;
        }
        else
        {
            isUser();
        }

        /*Intent intent = new Intent(Login.this,sidebar.class);
        startActivity(intent);*/
    }
});
}

private void isUser() {
    String userEnteredUsername =
    username.getText().getText().toString().trim();
    String userEnteredPassword =
    password.getText().getText().toString().trim();
    DatabaseReference reference=
    FirebaseDatabase.getInstance().getReference("users");
    Query checkUser =
    reference.orderByChild("phone").equalTo(userEnteredUsername);
    checkUser.addListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            if (snapshot.exists())
            {
                username.setError(null);
                username.setErrorEnabled(false);
                String passwordFromDB =

```



```

snapshot.child(userEnteredUsername).child("password").getValue(String.class);
        //System.out.println(passwordFromDB);
        // if(passwordFromDB.equals(userEnteredPassword))
        if(userEnteredPassword.equals(passwordFromDB))
        {
            password.setError(null);
            password.setErrorEnabled(false);
            String nameFromDB =
snapshot.child(userEnteredUsername).child("name").getValue(String.class);
            String usernameFromDB =
snapshot.child(userEnteredUsername).child("username").getValue(String.class);
            String phoneNoFromDB =
snapshot.child(userEnteredUsername).child("phone").getValue(String.class);
            String emailFromDB =
snapshot.child(userEnteredUsername).child("email").getValue(String.class);

            /*SharedPreferences sharedPref =
getSharedPreferences("myKey", MODE_PRIVATE);
            SharedPreferences.Editor editor = sharedPref.edit();
            editor.putString("value", phoneNoFromDB);
            editor.apply();*/

            Intent intent = new
Intent(getApplicationContext(), sidebar.class);
            //Intent inte = new
Intent(getApplicationContext(), dummy.class);
            //inte.putExtra("email", emailFromDB);
            intent.putExtra("email", emailFromDB);
            intent.putExtra("name", nameFromDB);
            intent.putExtra("password", passwordFromDB);
            intent.putExtra("phone", phoneNoFromDB);
            intent.putExtra("username", usernameFromDB);
            //
            intent.putExtra("Flag", "1");
            startActivity(intent);
        }
        else
        {
            password.setError("wrong password");
            password.requestFocus();
        }

    }
    else
    {
        username.setError("no such user exist");
        username.requestFocus();
    }
}
@Override
public void onCancelled(@NonNull DatabaseError error) {
}
});
}
private Boolean validateUsername()
{

```

```

        String val = username.getText().toString();
        if (val.isEmpty())
        {
            username.setError("Field can not be empty");
            return false;
        }
        else
        {
            username.setError(null);
            username.setErrorEnabled(false);
            return true;
        }
    }
    private Boolean validatePassword()
    {
        String val = password.getText().toString();
        if (val.isEmpty())
        {
            password.setError("Field can not be empty");
            return false;
        }
        else
        {
            password.setError(null);
            password.setErrorEnabled(false);
            return true;
        }
    }
}
}

```

MainActivity.java

```

package com.example.book_holic;
import android.app.ActivityOptions;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.util.Pair;
import android.view.View;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private static int SPLASH_SCREEN = 2000;
    //
    Animation topanim,bottomanim;
    ImageView image;
}

```

```

    TextView logo;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.activity_main);

        topanim = AnimationUtils.loadAnimation(this,R.anim.top_animation);
        bottomanim = AnimationUtils.loadAnimation(this,R.anim.bottom_animation);

        image = findViewById(R.id.logo);
        logo = findViewById(R.id.textView2);
        image.setAnimation(topanim);
        logo.setAnimation(bottomanim);

        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent intent = new Intent(MainActivity.this, sidebarNewUser.class);
                Pair[] pairs = new Pair[2];

                pairs[0] = new Pair<View,String>(image,"logo_name");
                pairs[1] = new Pair<View,String>(logo,"logo_text");
                ActivityOptions options =
ActivityOptions.makeSceneTransitionAnimation(MainActivity.this,pairs);
                startActivity(intent,options.toBundle());
                finish();
            }
        },SPLASH_SCREEN);
    }
}

```

Model.java

```

package com.example.book_holic;

public class Model {
    private String imageUrl;

    public Model() {
    }
    public Model(String imageUrl) {
        this.imageUrl = imageUrl;
    }

    public String getImageUrl() {
        return imageUrl;
    }

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

```

MyAdapter.java

```
package com.example.book_holic;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Filter;
import android.widget.Filterable;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

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

import com.bumptech.glide.Glide;
import com.firebase.ui.database.FirebaseRecyclerOptions;

import org.jetbrains.annotations.NotNull;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collection;
import java.util.List;

import static com.firebase.ui.auth.AuthUI.getApplicationContext;

public class MyAdapter extends RecyclerView.Adapter<MyAdapter.MyViewHolder>
implements Filterable {

    ArrayList<bookdata> list;
    ArrayList<bookdata> listFull;
    Context context;

    public MyAdapter(ArrayList<bookdata> list, Context context) {
        this.list = list;
        this.context = context;
        // listFull = new ArrayList<>(list);
        listFull =list;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View v=
        LayoutInflater.from(context).inflate(R.layout.recycler_item,parent,false);
        return new MyViewHolder(v);
    }
}
```

```

    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
        bookdata book=list.get(position);
//
Glide.with(context).load(list.get(position).getImageUrl()).into(holder.imageView);
        Glide.with(context).load(book.getImageUrl()).into(holder.imageView);

        holder.title.setText(book.getTitle());
        holder.desc.setText(book.getDescription());
        holder.category.setText(book.getCategory());
        holder.for_s_e.setText(book.getType());
        holder.price.setText(book.getAmount());
        holder.phone.setText(book.getphone());

        holder.itemView.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("RestrictedApi")
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(context,contact.class);
                //intent.putExtra("imageurl", book.hashCode());
                intent.putExtra("imageurl", book.getImageUrl());
                intent.putExtra("title", book.getTitle());
                intent.putExtra("description", book.getDescription());
                intent.putExtra("category", book.getCategory());
                intent.putExtra("for", book.getType());
                intent.putExtra("price", book.getAmount());
                intent.putExtra("phone", book.getphone());
                context.startActivity(intent);
            }
        });
    }
}

public static class MyViewHolder extends RecyclerView.ViewHolder{

    TextView title,desc,category,for_s_e,price,phone,phoneTitle,descTitle;
    ImageView imageView;
    public MyViewHolder(@NonNull View itemView) {
        super(itemView);
        phoneTitle=itemView.findViewById(R.id.LblPhone);
        descTitle=itemView.findViewById(R.id.LblDesc);

        imageView = itemView.findViewById(R.id.bookLogo);
        title=itemView.findViewById(R.id.title1);
        desc=itemView.findViewById(R.id.desc1);
        category=itemView.findViewById(R.id.for1);
        for_s_e=itemView.findViewById(R.id.category1);
        price=itemView.findViewById(R.id.price1);

        phone=itemView.findViewById(R.id.Phone1);

        phone.setVisibility(View.GONE);
        phoneTitle.setVisibility(View.GONE);
    }
}

```

```

        desc.setVisibility(View.GONE);
        descTitle.setVisibility(View.GONE);
    }
}

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

@Override
public Filter getFilter() {
    return FilterUser;
}
private Filter FilterUser = new Filter() {

    @Override
    protected FilterResults performFiltering(CharSequence charSequence) {

        //String searchText = charSequence.toString().toLowerCase();
        ArrayList<bookdata>tempList = new ArrayList<>();
        if(charSequence.toString().trim().isEmpty())
        {

            Intent intent1 = new Intent(context,sidebarNewUser.class);
            tempList.addAll(listFull);
            Log.d("tempList", String.valueOf(tempList.get(0)));
            intent1.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
            context.startActivity(intent1);
        }
        else
        {
            for(bookdata item : listFull)
            {
                if(item.getCategory().toString().toLowerCase().contains(charSequence.toString().toLowerCase()) ||
                item.getTitle().toString().toLowerCase().contains(charSequence.toString().toLowerCase())){
                    tempList.add(item);
                }
            }
            FilterResults results = new FilterResults();
            results.values=tempList;
            return results;
        }
    }

    @Override
    protected void publishResults(CharSequence constraint, @NotNull FilterResults
results) {
        list.clear();
        list.addAll((ArrayList<bookdata>)results.values);
    }
}

```

```

        //list.addAll((Collection<? extends bookdata>) filterResults.values);
        notifyDataSetChanged();
    }
};
}

```

MyAdapterLogin.java

```

package com.example.book_holic;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Filter;
import android.widget.Filterable;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

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

import com.bumptech.glide.Glide;
import com.firebase.ui.database.FirebaseRecyclerOptions;

import org.jetbrains.annotations.NotNull;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collection;
import java.util.List;

import static com.firebase.ui.auth.AuthUI.getApplicationContext;

public class MyAdapterLogin extends RecyclerView.Adapter<MyAdapterLogin.MyViewHolder>
implements Filterable {

    ArrayList<bookdata> list;
    ArrayList<bookdata> listFull;
    Context context;

    public MyAdapterLogin(ArrayList<bookdata> list, Context context) {

        this.list = list;
        this.context = context;
        // listFull = new ArrayList<>(list);
        listFull =list;
    }

```

```

        @NonNull
        @Override

        public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
            View v=
            LayoutInflater.from(context).inflate(R.layout.recycler_item,parent,false);
            return new MyViewHolder(v);
        }

        @Override
        public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
            bookdata book=list.get(position);
            //
            Glide.with(context).load(list.get(position).getImageUrl()).into(holder.imageView);
            Glide.with(context).load(book.getImageUrl()).into(holder.imageView);

            holder.title.setText(book.getTitle());
            holder.desc.setText(book.getDescription());
            holder.category.setText(book.getCategory());
            holder.for_s_e.setText(book.getType());
            holder.price.setText(book.getAmount());
            holder.phone.setText(book.getphone());

            holder.itemView.setOnClickListener(new View.OnClickListener() {
                @SuppressWarnings("RestrictedApi")
                @Override
                public void onClick(View v) {
                    Intent intent = new Intent(context,contact.class);
                    //intent.putExtra("imageurl", book.hashCode());
                    intent.putExtra("imageurl", book.getImageUrl());
                    intent.putExtra("title", book.getTitle());
                    intent.putExtra("description", book.getDescription());
                    intent.putExtra("category", book.getCategory());
                    intent.putExtra("for", book.getType());
                    intent.putExtra("price", book.getAmount());
                    intent.putExtra("phone", book.getphone());
                    context.startActivity(intent);
                }
            });
        }

        public static class MyViewHolder extends RecyclerView.ViewHolder{

            TextView title,desc,category,for_s_e,price,phone,phoneTitle,descTitle;
            ImageView imageView;
            public MyViewHolder(@NonNull View itemView) {
                super(itemView);
                phoneTitle=itemView.findViewById(R.id.lblPhone);
                descTitle=itemView.findViewById(R.id.lblDesc);

                imageView = itemView.findViewById(R.id.bookLogo);
                title=itemView.findViewById(R.id.title1);
                desc=itemView.findViewById(R.id.desc1);
            }
        }
    }

```



```

        category=itemView.findViewById(R.id.for1);
        for_s_e=itemView.findViewById(R.id.category1);
        price=itemView.findViewById(R.id.price1);

        phone=itemView.findViewById(R.id.Phone1);

        phone.setVisibility(View.GONE);
        phoneTitle.setVisibility(View.GONE);
        desc.setVisibility(View.GONE);
        descTitle.setVisibility(View.GONE);
    }
}

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

@Override
public Filter getFilter() {
    return FilterUser;
}
private Filter FilterUser = new Filter() {

    @Override
    protected FilterResults performFiltering(CharSequence charSequence) {

        //String searchText = charSequence.toString().toLowerCase();
        ArrayList<bookdata>tempList = new ArrayList<>();
        if(charSequence.toString().trim().isEmpty())
        {

            Intent intent1 = new Intent(context,sidebar.class);
            tempList.addAll(listFull);
            Log.d("tempList", String.valueOf(tempList.get(0)));
            intent1.addFlags(Intent.FLAG_ACTIVITY_NO_ANIMATION);
            context.startActivity(intent1);
        }
        else
        {
            for(bookdata item : listFull)
            {
                if(item.getCategory().toString().toLowerCase().contains(charSequence.toString().toLowerCase()) ||
                item.getTitle().toString().toLowerCase().contains(charSequence.toString().toLowerCase())){
                    tempList.add(item);
                }
            }
        }
        FilterResults results = new FilterResults();
        results.values=tempList;
    }
}

```

```

        return results;
    }

    @Override
    protected void publishResults(CharSequence constraint, @NotNull FilterResults
results) {
        list.clear();
        list.addAll((ArrayList<bookdata>)results.values);
        //list.addAll((Collection<? extends bookdata>) filterResults.values);
        notifyDataSetChanged();
    }
};
}

```

MyBooks.java

```

package com.example.book_holic;

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

import android.os.Bundle;
import android.widget.Toast;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class MyBooks extends AppCompatActivity {
    //recycler view
    String phone;
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    DatabaseReference databaseReference1;
    RemoveAdapter myAdapter;
    ArrayList<bookdata> list;

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

        phone = getIntent().getStringExtra("phone");
        recyclerView = findViewById(R.id.recyclerView);
        databaseReference = FirebaseDatabase.getInstance().getReference();
        databaseReference1 = databaseReference.child("book");
    }
}

```

```

recyclerView.setHasFixedSize(true);
recyclerView.setLayoutManager(new LinearLayoutManager(this));
list = new ArrayList<>();
myAdapter = new RemoveAdapter(this, list);
recyclerView.setAdapter(myAdapter);

databaseReference1.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        for (DataSnapshot ds : dataSnapshot.getChildren()) {
            String key = ds.getKey();
            if (key.equals(phone)) {

                //Log.d("check",key);
                DatabaseReference subKey = databaseReference1.child(key);
                ValueEventListener childEventListener =
subKey.addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull DataSnapshot snapshot) {
                        for (DataSnapshot ds1 : snapshot.getChildren()) {
                            bookdata book = ds1.getValue(bookdata.class);
                            list.add(book);
                        }
                        myAdapter.notifyDataSetChanged();
                    }

                    @Override
                    public void onCancelled(@NonNull DatabaseError error) {

                    }

                });
            }
        }
        // else{
        //     Toast.makeText(getApplicationContext(),"book not
found",Toast.LENGTH_SHORT).show();
        // }
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }
});
}
}

```

Recycleview.java

```

package com.example.book_holic;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import androidx.annotation.NonNull;

```

```

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
public class recycleview extends AppCompatActivity {

    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    MyAdapter myAdapter;
    ArrayList<bookdata> list;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_recycleview);
        recyclerView = findViewById(R.id.recyclerView);
        databaseReference = FirebaseDatabase.getInstance().getReference("book");
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        list = new ArrayList<>();
        myAdapter = new MyAdapter(list,this);
        recyclerView.setAdapter(myAdapter);
        databaseReference.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                for (DataSnapshot dataSnapshot: snapshot.getChildren()){
                    bookdata book = dataSnapshot.getValue(bookdata.class);
                    list.add(book);
                }
                myAdapter.notifyDataSetChanged();
            }
            @Override
            public void onCancelled(@NonNull DatabaseError error) {

            }
        });
    }
}

```

RemoveAdapter.java

```

package com.example.book_holic;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;

```

```

import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

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

import com.bumptech.glide.Glide;

import java.util.ArrayList;

import static com.firebase.ui.auth.AuthUI.getApplicationContext;

public class RemoveAdapter extends RecyclerView.Adapter<RemoveAdapter.MyViewHolder> {

    Context context;
    ArrayList<bookdata> list;

    public RemoveAdapter(Context context, ArrayList<bookdata> list) {
        this.context = context;
        this.list = list;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View v =
LayoutInflater.from(context).inflate(R.layout.recycler_item,parent,false);
        return new MyViewHolder(v);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, int position) {
        bookdata book=list.get(position);
        //
        Glide.with(context).load(list.get(position).getImageUrl()).into(holder.imageView);
        Glide.with(context).load(book.getImageUrl()).into(holder.imageView);

        holder.title.setText(book.getTitle());
        holder.desc.setText(book.getDescription());
        holder.category.setText(book.getCategory());
        holder.for_s_e.setText(book.getType());
        holder.price.setText(book.getAmount());
        holder.phone.setText(book.getphone());
        holder.itemView.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("RestrictedApi")
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(context,RemoveBook.class);
                intent.putExtra("imageurl", book.getImageUrl());
                intent.putExtra("title", book.getTitle());
            }
        });
    }
}

```

```

        intent.putExtra("description", book.getDescription());
        intent.putExtra("category", book.getCategory());
        intent.putExtra("for", book.getType());
        intent.putExtra("price", book.getAmount());
        intent.putExtra("phone", book.getphone());
        context.startActivity(intent);
    }
});
}

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

public static class MyViewHolder extends RecyclerView.ViewHolder{

    TextView title,desc,category,for_s_e,price,phone,phoneTitle,descTitle;
    ImageView imageView;
    public MyViewHolder(@NonNull View itemView) {
        super(itemView);
        phoneTitle=itemView.findViewById(R.id.LblPhone);
        descTitle=itemView.findViewById(R.id.LblDesc);

        imageView = itemView.findViewById(R.id.bookLogo);
        title=itemView.findViewById(R.id.title1);
        desc=itemView.findViewById(R.id.desc1);
        category=itemView.findViewById(R.id.for1);
        for_s_e=itemView.findViewById(R.id.category1);
        price=itemView.findViewById(R.id.price1);
        phone=itemView.findViewById(R.id.Phone1);

        phone.setVisibility(View.GONE);
        phoneTitle.setVisibility(View.GONE);
        desc.setVisibility(View.GONE);
        descTitle.setVisibility(View.GONE);

    }
}
}

```

RemoveBook.java

```

package com.example.book_holic;

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

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

```

```

import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class RemoveBook extends AppCompatActivity {
    String imageUrl,booktitle,bookDesc,bookCategory,bookFor,bookPrice,phone;
    ImageView _bookImage;
    Button _remove;
    TextView _bookTitle,_bookDesc,_bookCategory,_bookFor,_bookPrice;
    DatabaseReference databaseReference;
    DatabaseReference databaseReference1;
    ArrayList<bookdata> list;
    RemoveAdapter myAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_remove_book);
        //Integer imageUrl = getIntent().getIntExtra("imageUrl",0);
        imageUrl = getIntent().getStringExtra("imageUrl");
        booktitle = getIntent().getStringExtra("title");
        bookDesc = getIntent().getStringExtra("description");
        bookCategory = getIntent().getStringExtra("category");
        bookFor = getIntent().getStringExtra("for");
        bookPrice = getIntent().getStringExtra("price");
        phone = getIntent().getStringExtra("phone");

        _bookImage = findViewById(R.id.book_image);
        _bookTitle = findViewById(R.id.book_title);
        _bookDesc = findViewById(R.id.book_description);
        _bookCategory = findViewById(R.id.book_category);
        _bookFor = findViewById(R.id.book_for);
        _bookPrice = findViewById(R.id.book_price);
        _remove = findViewById(R.id.remove);

        databaseReference = FirebaseDatabase.getInstance().getReference();
        databaseReference1 = databaseReference.child("book");
        list = new ArrayList<>();
        myAdapter = new RemoveAdapter(this, list);

        // Glide.with(context).load(book.getImageUrl()).into(holder.imageView);
        Glide.with(this).asBitmap().load(imageUrl).into(_bookImage);

        _bookTitle.setText(booktitle);
        _bookDesc.setText(bookDesc);
        _bookCategory.setText(bookCategory);
        _bookFor.setText(bookFor);
    }
}

```

```

        _bookPrice.setText(bookPrice);

        _remove.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                databaseReference1.addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(DataSnapshot dataSnapshot) {
                        for (DataSnapshot ds : dataSnapshot.getChildren()) {
                            String key = ds.getKey();
                            if (key.equals(phone)) {

                                DatabaseReference subKey = databaseReference1.child(key);

                                ValueEventListener childEventListener =
subKey.addValueEventListener(new ValueEventListener() {
                                    @Override
                                    public void onDataChange(@NonNull DataSnapshot
snapshot) {

                                        for (DataSnapshot ds1 : snapshot.getChildren()) {
                                            bookdata book = ds1.getValue(bookdata.class);
                                            if (subKey.equals(booktitle)) {
                                                }
                                            ds1.getRef().removeValue();
                                        }
                                        myAdapter.notifyDataSetChanged();
                                    }
                                }

                                @Override
                                public void onCancelled(@NonNull DatabaseError error)
{

                                }
                            });
                        }
                    }
                }
            }
        });
    }
}
}
}

```

Sidebar.java

```

package com.example.book_holic;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;

```



```

import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.SearchView;
import android.widget.TextView;
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.cardview.widget.CardView;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.android.material.navigation.NavigationView;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class sidebar extends AppCompatActivity {
    NavigationView nav;
    View mHeaderView;

    //private MenuItem menuItem1;
    ActionBarDrawerToggle toggle;
    DrawerLayout drawerLayout;
    Toolbar toolbar;
    TextView fullname,about;

    //recycler view
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    DatabaseReference databaseReference1;
    MyAdapterLogin myAdapter;
    ArrayList<bookdata> list;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sidebar);

        //recycler view
        recyclerView = findViewById(R.id.recyclerView);
        databaseReference = FirebaseDatabase.getInstance().getReference();
        databaseReference1 = databaseReference.child("book");
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
    }

```

```

list = new ArrayList<>();
myAdapter = new MyAdapterLogin(list, this);
recyclerView.setAdapter(myAdapter);

databaseReference1.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        for (DataSnapshot ds : dataSnapshot.getChildren()) {
            String key = ds.getKey();
            //Log.d("check",key);
            DatabaseReference subKey = databaseReference1.child(key);
            ValueEventListener childEventListener =
subKey.addValueEventListener(new ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot snapshot) {
                    for (DataSnapshot ds1 : snapshot.getChildren()) {
                        bookdata book = ds1.getValue(bookdata.class);
                        list.add(book);
                        //Log.d("key",
String.valueOf(ds1.child("Title").getValue()));
                    }
                    myAdapter.notifyDataSetChanged();
                }

                @Override
                public void onCancelled(@NonNull DatabaseError error) {

                }
            });
        }
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }
});

toolbar = findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

nav = findViewById(R.id.navmenu);
mHeaderView = nav.getHeaderView(0);
fullname =(TextView) mHeaderView.findViewById(R.id.full_name);
about =(TextView) mHeaderView.findViewById(R.id.about);

String email = getIntent().getStringExtra("email");
String phone = getIntent().getStringExtra("phone");
String password = getIntent().getStringExtra("password");
String name = getIntent().getStringExtra("name");

about.setText(phone);

drawerLayout=findViewById(R.id.drawer);
toggle=new ActionBarDrawerToggle(this, drawerLayout, toolbar, R.string.open,
R.string.close);

```

```

toggle.getDrawerArrowDrawable().setColor(getResources().getColor(R.color.three));
drawerLayout.addDrawerListener(toggle);
toggle.syncState();

nav.setNavigationItemSelectedListener(new
NavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {

        switch (item.getItemId())
        {
            case R.id.menu_home :
                Intent home= new Intent(sidebar.this,sidebar.class);
                startActivity(home);
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
            case R.id.profile :
                Intent intent= new Intent(sidebar.this,UserProfile.class);
                intent.putExtra("phone",phone);
                intent.putExtra("email",email);
                intent.putExtra("name",name);
                intent.putExtra("password",password);
                startActivity(intent);
                //Toast.makeText(getApplicationContext(),"Profile panel is
open",Toast.LENGTH_LONG).show();
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
            case R.id.add_book :
                Intent i= new Intent(sidebar.this,addBook.class);
                i.putExtra("phone",phone);
                startActivity(i);
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
            case R.id.my_books :
                Intent myBooks= new Intent(sidebar.this,MyBooks.class);
                myBooks.putExtra("phone",phone);
                startActivity(myBooks);
                //Toast.makeText(getApplicationContext(),"My Books panel is
open",Toast.LENGTH_LONG).show();
                drawerLayout.closeDrawer(GravityCompat.START);
                break;

            case R.id.about :
                Intent abt=new Intent(sidebar.this,about.class);
                startActivity(abt);
                //Toast.makeText(getApplicationContext(),"Help panel is
open",Toast.LENGTH_LONG).show();
                drawerLayout.closeDrawer(GravityCompat.START);
                break;

            case R.id.logout :
                Intent inte= new Intent(sidebar.this,Login.class);
                startActivity(inte);

```

```

        Toast.makeText(getApplicationContext(),"Logout
Successfully",Toast.LENGTH_LONG).show();
        drawerLayout.closeDrawer(GravityCompat.START);
        break;
    }
    return true;
}
});
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.searchmenu,menu);
    MenuItem menuItem = menu.findItem(R.id.search);
    SearchView searchView = (SearchView)menuItem.getActionView();
    searchView.setBackgroundColor(getResources().getColor(R.color.three));
    searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String s) {
            return false;
        }

        @Override
        public boolean onQueryTextChange(String s) {
            myAdapter.getFilter().filter(s);
            return false;
        }
    });
    return super.onCreateOptionsMenu(menu);
}

@Override
public void onBackPressed() {
    AlertDialog.Builder builder=new AlertDialog.Builder(this);
    builder.setMessage("Logout!!..sure?").setCancelable(false)
        .setPositiveButton("yes", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                sidebar.super.onBackPressed();
            }
        }).setNegativeButton("No", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });
    AlertDialog alertDialog = builder.create();
    alertDialog.show();
}
}
}

```

SidebarNewUser.java

```
package com.example.book_holic;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.Drawable;
import android.os.Build;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;
import android.widget.SearchView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.core.content.ContextCompat;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

public class sidebarNewUser extends AppCompatActivity {
    NavigationView nav;
    View mHeaderView;

    ActionBarDrawerToggle toggle;
    DrawerLayout drawerLayout;
    Toolbar toolbar;

    //recyclerView
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    DatabaseReference databaseReference1;
    MyAdapter myAdapter, myAdapter1;
    ArrayList<bookdata> list;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_sidebar_new_user);
    //recycler view
    recyclerView = findViewById(R.id.recyclerView);
    databaseReference = FirebaseDatabase.getInstance().getReference();
    databaseReference1 = databaseReference.child("book");
    recyclerView.setHasFixedSize(true);
    recyclerView.setLayoutManager(new LinearLayoutManager(this));
    list = new ArrayList<>();
    myAdapter = new MyAdapter(list, this);

    recyclerView.setAdapter(myAdapter);
    //start fetching data
    //first iteration
    databaseReference1.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            for (DataSnapshot ds : dataSnapshot.getChildren()) {
                String key = ds.getKey();
                DatabaseReference subKey = databaseReference1.child(key);
                ValueEventListener childEventListener =
subKey.addValueEventListener(new ValueEventListener() {
                    @Override
                    public void onDataChange(@NonNull DataSnapshot snapshot) {
                        for (DataSnapshot ds1 : snapshot.getChildren()) {
                            bookdata book = ds1.getValue(bookdata.class);
                            list.add(book);
                            // Log.d(mi"key",
String.valueOf(ds1.child("Title").getValue()));
                        }
                        int i = myAdapter.getItemCount();
                        Log.d("adaptercount", String.valueOf(i));
                        myAdapter.notifyDataSetChanged();
                    }

                    @Override
                    public void onCancelled(@NonNull DatabaseError error) {

                }
            });
        }
    });

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }
});

toolbar = findViewById(R.id.toolbar1);
setSupportActionBar(toolbar);
nav = findViewById(R.id.navmenu1);
drawerLayout = findViewById(R.id.drawer1);
toggle = new ActionBarDrawerToggle(this, drawerLayout, toolbar,

```

```

R.string.open, R.string.close);

toggle.getDrawerArrowDrawable().setColor(getResources().getColor(R.color.three));
drawerLayout.addDrawerListener(toggle);
toggle.syncState();

    nav.setNavigationItemSelectedListener(new
NavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {

        switch (item.getItemId()) {
            case R.id.menu_home1:
                Intent home= new
Intent(sidebarNewUser.this,sidebarNewUser.class);
                startActivity(home);
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
            case R.id.about1:
                Intent abt = new Intent(sidebarNewUser.this, about.class);
                startActivity(abt);
                //Toast.makeText(getApplicationContext(),"Help panel is
open",Toast.LENGTH_LONG).show();
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
            case R.id.Login:
                Intent inte = new Intent(sidebarNewUser.this, Login.class);
                startActivity(inte);
                drawerLayout.closeDrawer(GravityCompat.START);
                break;
        }
        return true;
    }
});

}
@RequiresApi(api = Build.VERSION_CODES.M)
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.searchmenu,menu);
    MenuItem menuItem = menu.findItem(R.id.search);
    SearchView searchView = (SearchView)menuItem.getActionView();
    searchView.setBackgroundColor(getResources().getColor(R.color.three));
    //    ImageView searchIcon =
searchView.findViewById(android.support.v7.appcompat.R.id.search_button);
    //
searchIcon.setImageDrawable(ContextCompat.getDrawable(getActivity(),R.drawable.ic_sea
rch_icon));
    // ImageView searchIcon = menuItem.
searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
        @Override
        public boolean onQueryTextSubmit(String query) {
            //myAdapter.getFilter().filter(s.toLowerCase());
            return false;
        }
    }
}

```

```

        @Override
        public boolean onQueryTextChange(String s) {
            myAdapter.getFilter().filter(s);
            return false;
        }
    });
    return super.onCreateOptionsMenu(menu);
}

@Override
public void onBackPressed() {
    AlertDialog.Builder builder=new AlertDialog.Builder(this);
    builder.setMessage("Are you sure you want to Exit?").setCancelable(false)
        .setPositiveButton("yes", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                sidebarNewUser.super.onBackPressed();
            }
        }).setNegativeButton("No", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });
    AlertDialog alertDialog = builder.create();
    alertDialog.show();
}
}

```

SignUp.java

```

package com.example.book_holic;

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

import android.app.Dialog;
import android.content.Intent;
import android.net.wifi.hotspot2.pps.Credential;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.android.material.textfield.TextInputLayout;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthProvider;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;

```



```

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

import java.util.concurrent.TimeUnit;

public class SignUp extends AppCompatActivity {
    TextInputLayout regName,regUsername,regEmail,regPhone,regPassword;
    Button regBtn, regToLogin;
    private static final String key="SignUp";
    FirebaseDatabase rootNode;
    DatabaseReference reference;
    int flag = 0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sign_up);

        regName = findViewById(R.id.name);
        regUsername = findViewById(R.id.username);
        regEmail = findViewById(R.id.email);
        regPhone = findViewById(R.id.phone);
        regPassword = findViewById(R.id.password);
        regBtn = findViewById(R.id.go);
        regToLogin = findViewById(R.id.goToLogin);

        regBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(!validateName())
                { return; }
                else if(!validateUserName())
                { return;}
                else if(!validateEmail())
                { return; }
                else if(!validatePhoneNo())
                {return;}
                else {
                    isUser();
                }
            }
        });
        regToLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intent = new Intent(SignUp.this,Login.class);
                startActivity(intent);
            }
        });
    }
}

```

```
//validation
private Boolean validateName()
{
    String val = regName.getText().toString();
    if (val.isEmpty())
    {
        regName.setError("Field can not be empty");
        return false;
    }
    else
    {
        regName.setError(null);
        regName.setErrorEnabled(false);
        return true;
    }
}

private Boolean validateUserName()
{
    String val = regUsername.getText().toString();
    String noWhitespace = "\\A\\w{4,20}\\z";
    if (val.isEmpty())
    {
        regUsername.setError("Field can not be empty");
        return false;
    }

    else
    {
        regUsername.setError(null);
        regUsername.setErrorEnabled(false);
        return true;
    }
}

private Boolean validateEmail()
{
    String val = regEmail.getText().toString();
    String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";
    if (val.isEmpty())
    {
        regEmail.setError("Field can not be empty");
        return false;
    }
    else if(!val.matches(emailPattern))
    {
        regEmail.setError("Invalid Email");
        return false;
    }
    else
    {
        regEmail.setError(null);
        regEmail.setErrorEnabled(false);
    }
}
```

```

        return true;
    }
}

private Boolean validatePhoneNo()
{
    String val = regPhone.getEditText().getText().toString();
    if (val.isEmpty())
    {
        regPhone.setError("Field can not be empty");
        return false;
    }
    else
    {
        regPhone.setError(null);
        regPhone.setErrorEnabled(false);
        return true;
    }
}

private void isUser() {
    String userEnteredUsername =
regPhone.getEditText().getText().toString().trim();
    DatabaseReference reference=
FirebaseDatabase.getInstance().getReference("users");
    Query checkUser =
reference.orderByChild("phone").equalTo(userEnteredUsername);
    checkUser.addListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            if (snapshot.exists())
            {
                regPhone.setError("user exist");
                regPhone.requestFocus();
            }
            else
            {
                regPhone.setError(null);
                regPhone.setErrorEnabled(false);
                // rootNode = FirebaseDatabase.getInstance();
                // reference = rootNode.getReference("users");
                String name = regName.getEditText().getText().toString();
                String username = regUsername.getEditText().getText().toString();
                String email = regEmail.getEditText().getText().toString();
                String phoneNo = regPhone.getEditText().getText().toString();
                String password = regPassword.getEditText().getText().toString();

                Intent intent = new
Intent(getApplicationContext(),VerifyPhoneNo.class);
                intent.putExtra("flag", flag);
                intent.putExtra("name",name);
                intent.putExtra("username",username);
                intent.putExtra("email",email);
                intent.putExtra("phoneNo",phoneNo);

```

```

        intent.putExtra("password",password);

        startActivity(intent);
    }
}
@Override
public void onCancelled(@NonNull DatabaseError error) {
}
});
}
// phone number verification
/* public void verifyphonenumber(String regPhone) {

    PhoneAuthProvider.getInstance().verifyPhoneNumber("+91" + regPhone, 60,
    TimeUnit.SECONDS, SignUp.this,
        new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {
            @Override
            public void onVerificationCompleted(@NonNull PhoneAuthCredential
phoneAuthCredential) {

                signInUser(phoneAuthCredential);
            }

            @Override
            public void onVerificationFailed(@NonNull FirebaseException e) {
                Log.d(key, "On varification failed" +
e.getLocalizedMessage());
            }

            @Override
            public void onCodeSent(@NonNull String varificationId, @NonNull
PhoneAuthProvider.ForceResendingToken forceResendingToken) {
                super.onCodeSent(varificationId, forceResendingToken);
                Log.d("key1",regPhone);
                Dialog dialog = new Dialog(SignUp.this);
                dialog.setContentview(R.layout.verify_popup);
                EditText vc = dialog.findViewById(R.id.OTP);
                Button vb = dialog.findViewById(R.id.otpCheck);
                vb.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        String verificationCode = vc.getText().toString();
                        if (varificationId.isEmpty()) {
                            return;
                        }
                        PhoneAuthCredential credential =
PhoneAuthProvider.getCredential(varificationId, verificationCode);
                        signInUser(credential);
                    }
                });
                dialog.show();
            }
        });
}
});
});

```

```

    }
    private void signInUser(PhoneAuthCredential credential) {

FirebaseAuth.getInstance().signInWithCredential(credential).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        if (task.isSuccessful()) {
            startActivity(new Intent(SignUp.this, Login.class));
        } else {
            Log.d(key, "On complete" +
task.getException().getLocalizedMessage());
        }
    }
});

}

/*private Boolean validatePassword()
{
    String val = regPassword.getText().toString();
    String passwordVal = "^" +
        "(?=.*[0-9])" + //at least one digit
        "(?=.*[a-z])" + //at least one lower case letter
        "(?=.*[A-Z])" + //at least one upper case letter
        "(?=.*[a-zA-Z])" + //any letters
        "(?=.*[@#$%^&+=])" + //at least one special character
        "(?=\s+$)" + // no white space
        ".{4,}" + // at least four character
        "$";
    if (val.isEmpty())
    {
        regPassword.setError("Field can not be empty");
        return false;
    }
    else if(!val.matches(passwordVal))
    {
        regPassword.setError("password is too weak");
        return false;
    }
    else
    {
        regPassword.setError(null);
        regPassword.setEnabled(false);
        return true;
    }
}*/

//save data in the firebase on button click
/* public void registerUser(View view){
    if (!validateName() ) {
        return;
    }
    rootNode =FirebaseDatabase.getInstance();
    reference=rootNode.getReference("users");
}

```

```

        String name = regName.getEditText().getText().toString();
        String username = regUsername.getEditText().getText().toString();
        String email = regEmail.getEditText().getText().toString();
        String phoneNo = regPhone.getEditText().getText().toString();
        String password = regPassword.getEditText().getText().toString();

        UserHelper helperClass = new
UserHelper(name,username,email,phoneNo,password);
        reference.child(phoneNo).setValue(helperClass);
    }*/
}

```

UserHelper.java

```

package com.example.book_holic;

import android.content.Intent;

public class UserHelper {
    String name,username,email,phone,password;
    public UserHelper() {

    }

    public UserHelper(String name, String username, String email, String phone,
String password) {
        this.name = name;
        this.username = username;
        this.email = email;
        this.phone = phone;
        this.password = password;

    }

    public String getName() {
        return name;
    }

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

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getEmail() {
        return email;
    }
}

```

```

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

    public String getPhone() {
        return phone;
    }

    public void setPhone(String phone) {
        this.phone = phone;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}

```

UserProfile.java

```

package com.example.book_holic;

import androidx.appcompat.app.AppCompatActivity;

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

import com.google.android.material.textfield.TextInputLayout;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

public class UserProfile extends AppCompatActivity {
    TextInputLayout full_name,email,password;
    String phoneNo;
    Button update;
    String _fullname,_email,_password,_phoneno;
    DatabaseReference reference;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user_profile);

        reference = FirebaseDatabase.getInstance().getReference("users");
        update = findViewById(R.id.update);
        full_name = findViewById(R.id.full_name);
        email = findViewById(R.id.email);
    }
}

```

```

        password = findViewById(R.id.password);
        showAllUserData();
        update.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                update(v);
            }
        });
    }
    private void showAllUserData()
    {
        Intent intent = getIntent();
        _fullname = intent.getStringExtra("name");
        _email = intent.getStringExtra("email");
        _password = intent.getStringExtra("password");
        _phoneno = intent.getStringExtra("phone");

        full_name.getEditText().setText(_fullname);
        email.getEditText().setText(_email);
        password.getEditText().setText(_password);
    }

    public void update(View view){
        if(isPasswordChanged() | isNameChanged() | isEmailChanged())
        {
            Toast.makeText(this,"Data has been updated",Toast.LENGTH_LONG).show();
        }
        else
        {
            Toast.makeText(this,"Data is same and can not be updated",Toast.LENGTH_LONG).show();
        }
    }

    private boolean isPasswordChanged() {
        if(!_password.equals(password.getEditText().getText().toString()))
        {
            reference.child(_phoneno).child("password").setValue(password.getEditText().getText().toString());
            _password = password.getEditText().getText().toString();
            return true;
        }
        else
        {
            return false;
        }
    }
    private boolean isNameChanged()
    {
        if(!_fullname.equals(full_name.getEditText().getText().toString()))
        {

```



```

reference.child(_phoneno).child("name").setValue(full_name.getText().toString());
        _fullname = full_name.getText().toString();
        return true;
    }
    else
    {
        return false;
    }
}
private boolean isEmailChanged() {
    if(!_email.equals(email.getText().toString()))
    {
reference.child(_phoneno).child("email").setValue(email.getText().toString());
        _email = email.getText().toString();
        return true;
    }
    else
    {
        return false;
    }
}
}
}

```

VerifyPhoneNo.java

```

package com.example.book_holic;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.arch.core.executor.TaskExecutor;

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

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.android.gms.tasks.TaskExecutors;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthOptions;
import com.google.firebase.auth.PhoneAuthProvider;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

```

```

import java.util.concurrent.TimeUnit;

public class VerifyPhoneNo extends AppCompatActivity {
    String verificationCodeBySystem;
    Button verify_btn;
    EditText phoneNoEnteredByTheUser;
    ProgressBar progressBar;
    FirebaseAuth mAuth;
    DatabaseReference reference;
    FirebaseDatabase rootNode;
    String name,username,email,phoneNo,password;
    public int flag;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_verify_phone_no);
        verify_btn = findViewById(R.id.verify_btn);
        phoneNoEnteredByTheUser =
findViewById(R.id.verification_code_entered_by_user);
        progressBar = findViewById(R.id.progress_bar);
        mAuth = FirebaseAuth.getInstance();
        flag=Integer.parseInt(getIntent().getExtras().get("flag").toString());
        phoneNo = getIntent().getStringExtra("phoneNo");

        if(flag==0) {
            password = getIntent().getStringExtra("password");
            name = getIntent().getStringExtra("name");
            username = getIntent().getStringExtra("username");
            email = getIntent().getStringExtra("email");
        }
        rootNode = FirebaseDatabase.getInstance();
        reference = rootNode.getReference("users");
        sendVerificationCodeToUser(phoneNo);
        progressBar.setVisibility(View.GONE);
        verify_btn.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                String code = phoneNoEnteredByTheUser.getText().toString();
                if(code.isEmpty() || code.length()<6)
                {
                    phoneNoEnteredByTheUser.setError("wrong otp");
                    phoneNoEnteredByTheUser.requestFocus();
                    return;
                }
                progressBar.setVisibility(View.VISIBLE);
                verifyCode(code);
            }
        });
    }

    //oncreate ends

    private void sendVerificationCodeToUser(String phoneNo) {

```

```

        PhoneAuthOptions options =
            PhoneAuthOptions.newBuilder(mAuth)
                .setPhoneNumber("+91" + phoneNo)           // Phone number to
verify
                .setTimeout(60L, TimeUnit.SECONDS) // Timeout and unit
                .setActivity(this)                     // Activity (for callback
binding)
                .setCallbacks(mCallbacks)              //
OnVerificationStateChangedCallbacks
                .build();
        PhoneAuthProvider.verifyPhoneNumber(options);

    }
    //mCallbacks starts
    private PhoneAuthProvider.OnVerificationStateChangedCallbacks mCallbacks = new
PhoneAuthProvider.OnVerificationStateChangedCallbacks() {

        @Override
        public void onCodeSent(@NonNull String s, @NonNull
PhoneAuthProvider.ForceResendingToken forceResendingToken) {
            super.onCodeSent(s, forceResendingToken);
            verificationCodeBySystem = s;
        }

        @Override
        public void onVerificationCompleted(@NonNull PhoneAuthCredential
phoneAuthCredential) {
            String code = phoneAuthCredential.getSmsCode();
            if(code!=null)
            {
                progressBar.setVisibility(View.VISIBLE);
                verifyCode(code);
            }
        }

        @Override
        public void onVerificationFailed(@NonNull FirebaseException e) {
            Toast.makeText(VerifyPhoneNo.this,e.getMessage(),
Toast.LENGTH_SHORT).show();
        }

    };
    //mCallbacks ends

    private void verifyCode(String codeByUser){
        PhoneAuthCredential credential =
PhoneAuthProvider.getCredential(verificationCodeBySystem,codeByUser);
        signInTheUserByCredential(credential);
    }

    private void signInTheUserByCredential (PhoneAuthCredential credential){

```

```

mAuth.signInWithCredential(credential).addOnCompleteListener(VerifyPhoneNo.this, new
OnCompleteListener<AuthResult>() {

    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        if (flag==0)
        {
            if (task.isSuccessful()) {
                UserHelper helperClass = new UserHelper(name, username,
email, phoneNo, password);
                reference.child(phoneNo).setValue(helperClass);
                Intent intent = new Intent(getApplicationContext(),
Login.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
                startActivity(intent);
            }
            else {
                Toast.makeText(VerifyPhoneNo.this,
task.getException().getMessage(), Toast.LENGTH_SHORT).show();
            }
        }
        else
        {
            if (task.isSuccessful()) {
                Intent intent = new Intent(getApplicationContext(),
forgetPassword.class);
                intent.putExtra("phoneNo",phoneNo);
                //intent.putExtra("password",password);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
                startActivity(intent);
            }
            else {
                Toast.makeText(VerifyPhoneNo.this,
task.getException().getMessage(), Toast.LENGTH_SHORT).show();
            }
        }
    }
});
}
}

```

Build.gradle (Project: Book holic)

```

// Top-level build file where you can add configuration options common to all sub-
projects/modules.
buildscript {
    repositories {
        google()
        jcenter()
    }
}

```

```

    }
    dependencies {
        classpath "com.android.tools.build:gradle:4.1.2"
        classpath 'com.google.gms:google-services:4.3.5'

        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

allprojects {
    repositories {
        google()
        jcenter()
        maven {url "https://maven.google.com"}
        maven { url 'https://jitpack.io' }
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}

```

Build.gradle (Module: Book_holic.app)

```

plugins {
    id 'com.android.application'
    id 'com.google.gms.google-services'
}

android {
    compileSdkVersion 30
    buildToolsVersion "30.0.3"

    defaultConfig {
        applicationId "com.example.book_holic"
        minSdkVersion 21
        targetSdkVersion 30
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'),
'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

```

```

}
dependencies {

    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'com.google.android.material:material:1.3.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    implementation 'com.google.android.material:material:1.1.0'
    implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.3.0'
    implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.3.0'
    implementation 'com.google.firebase:firebase-storage:19.2.2'
    //noinspection GradleCompatible
    implementation 'com.firebaseui:firebase-ui-auth:6.4.0'
    //noinspection GradleCompatible
    implementation 'com.android.support:design:28.0.0'
    implementation 'com.google.firebase:firebase-database:19.7.0'
    implementation 'com.google.firebase:firebase-auth:20.0.4'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
    implementation 'com.firebaseui:firebase-ui-database:6.0.2'
    implementation 'com.tuyenmonkey:mkloader:1.4.0'
    //implementation "com.lhoyong:imagepicker:1.4.6"
    implementation 'com.myhexaville:smart-image-picker:1.0.4'
    implementation 'com.github.yalantis:ucrop:2.2.6'
    //implementation 'com.google.android.material:material:1.4.0-alpha01'
    implementation 'com.squareup.picasso:picasso:2.71828'
    implementation 'com.github.bumptech.glide:glide:4.12.0'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.12.0'
    // for sms varification
    implementation 'com.google.firebase:firebase-auth'
    implementation 'com.google.android.gms:play-services-auth:17.0.0'
    implementation 'com.google.android.gms:play-services-auth-api-phone:17.4.0'
    implementation platform('com.google.firebase:firebase-bom:27.1.0');
    implementation 'androidx.browser:browser:1.2.0'
    implementation 'com.google.android.gms:play-services-safetynet:17.0.0'
    implementation 'androidx.browser:browser:1.3.0'
}

```