



Pakistan Air Force
Karachi Institute of Economics & Technology

**A Responsive Book-Exchange-System application
Developed With Android Studio and Firebase**

**Submitted by: Javeria Asif (9442)
Javeria Hassan (9517)**

Advisor: Sir Ehtisham Khan

College of Computing & Information Sciences

A final year project report presented to the College of Computing & Information Sciences,
Pakistan Air Force – Karachi Institute of Economics & Technology in partial fulfilment of the requirements of the
degree of Bachelor of Science

2020-2021

Abstract

Online Book Exchange System is an android application which can be used to exchange books within a community. This system is developed with a purpose of helping users to sell or buy books. Any user can register to search for a book and obtain the owner's information and can also contact them via chat using the same application. The user can also put in a request online to buy or sell a book. This system is mainly intended to help the students exchange their course books within a particular area. Online book exchange system is a platform where students can sell or buy old books in half price.

Acknowledgements

First of all we would like to thanks The Almighty Allah for everything He blessed us with and without whom no endeavor including this one could have been successful then, all the teachers, institutes and mentors that have always supported us and helped us reach this level. Our most humble regards to the Director CoCIS, Khalid Khan, our Program Managers, Kashif Bashir and Sohail Imran, our DBMS professor yet again, Sir Sohail Imran, and the outstanding FYP committee members, Muhammad Yassen Khan, Ayub Latif and Dr. Affan Alim that given us time and again guided us, tested us and clarified many problematic aspects of our work. Adding to the long list we are grateful to have had Ehtisham Khan as our advisor. And saving the best for the last we would like to acknowledge our parents who supported us in our difficult situations.

Jaweria Asif (9442)

Javeria Hassan (9517)

January 2021

1. Introduction

Our project intends to provide users platform to sell their old books in reasonable price and let other user to purchase it. Any user can post their old books and any user who needed that book can buy it by contacting to the seller through chat while staying on the same application and also our application is going to recommend seller to the buyers. In addition users can post their queries related to studies and coding on discussion forum and any other user can answer that query once that query has been accepted by the admin as well as there also exist thought of the day in which there are multiple motivational quotes that acts as energizers for students. There exist options name as educational institute in which user can search nearby universities and colleges by applying different kind of filters. The user would get the result according to the percentage he/she has scored in matric and intermediate levels which let those selecting colleges and universities while sitting at home without physically visited that particular university or college. However student can also use it for finding their answers related to their queries and they can also get benefit by selling their old books and getting profit.

1.1. Motivations

Currently, there is limited number of ways for students to get textbooks for their classes. Some textbooks are very expensive; some are only available in pdf format. Student who needs hardcopy of a textbook but are not able to purchase them, because those books are very expensive. Some students spends exorbitant amount of money on textbooks which they may only use for a single term or a semester. The intended result is that students can help themselves by selling their old book by to other students who are studying the same course and need that book in hardcopy.

1.2. Objectives and Scope

The goal was to design and implement powerful, user-friendly, multi-user Book-Exchange-System developed using Android Studio, JAVA. It aims to be a fully functional and professional Book-Exchange-System which can be used by any user and especially If you are a student and you want to earn money, and you cannot do the job then this application provides you an option where you can sell your old course books and get money in exchange of them as well as if there is no one to give you information about which college or university is right for you then this application is going to assist you for that purpose as well.

The scope and goals kept in view are summarized as follows

- Easy tracking of users post
- Analyzing user queries
- Verification of user
- Tracking of user search activities
- Giving recommendation
- Tracking user Comments
- Crawling authentic websites

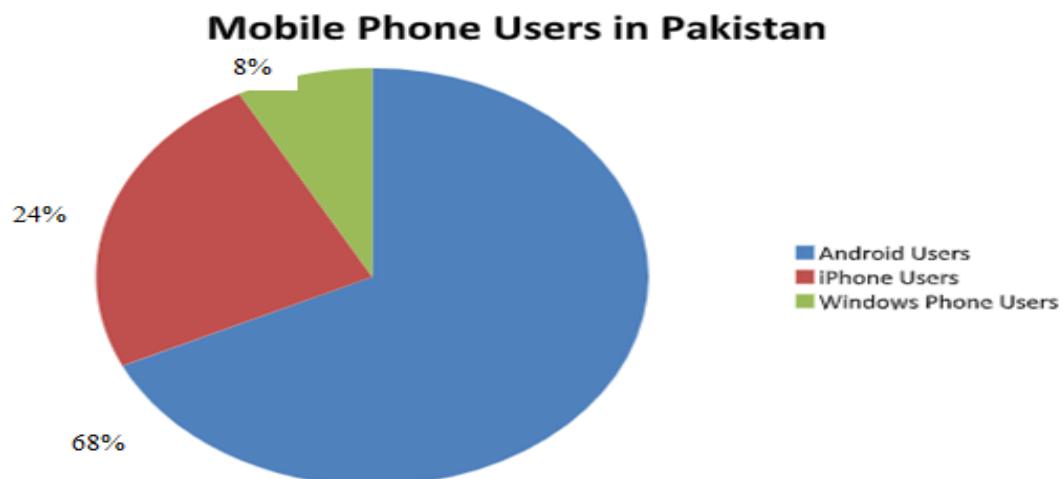
2.Methodology

In this we are going to introduce the methodology behind the Book-Exchange concept. One of the main goals of the project was to create something to WPI community could benefit from with the hope of ultimately expanding to other universities. We hope that someone may be able to benefit from the functionality of our app, or select to contribute the remaining milestones documented at the end of this paper. For many students there is no cost effective method of purchasing a textbook that they may only use one time. We sought to improve this process. There are many options for purchasing textbooks, none of which are financially compelling. With Book-Exchange, we wanted to offer low textbook prices within an even greater convenience than renting from Amazon. In order to accomplish this, we would need our own platform to foster a free market.

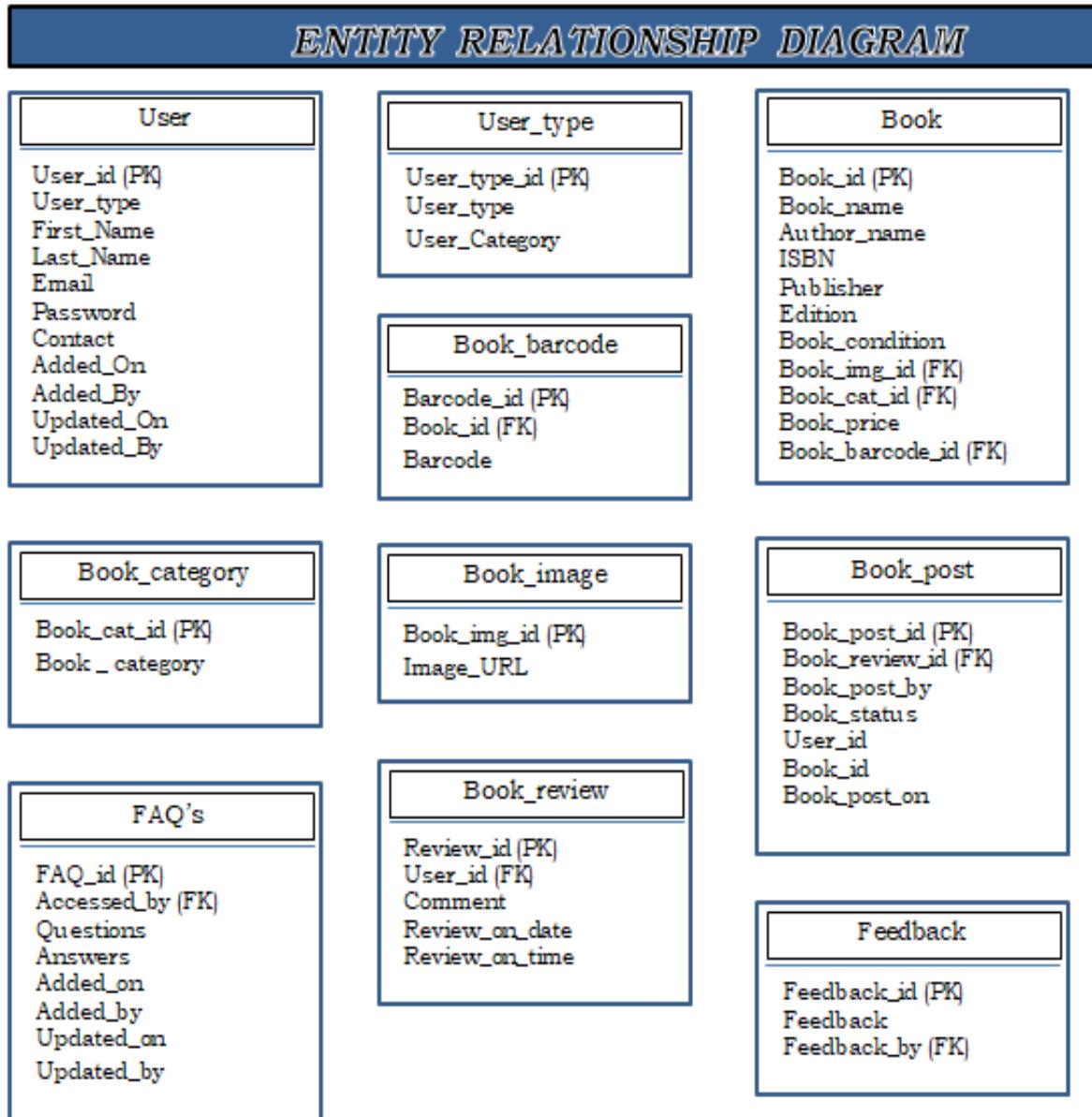
When planning this project, it was important to determine which platform would be best suit for this type of market place. Traditionally, these sorts of applications are web based. However, due to convenience and features such as image uploads, it was determined a mobile app will provide users the best experience

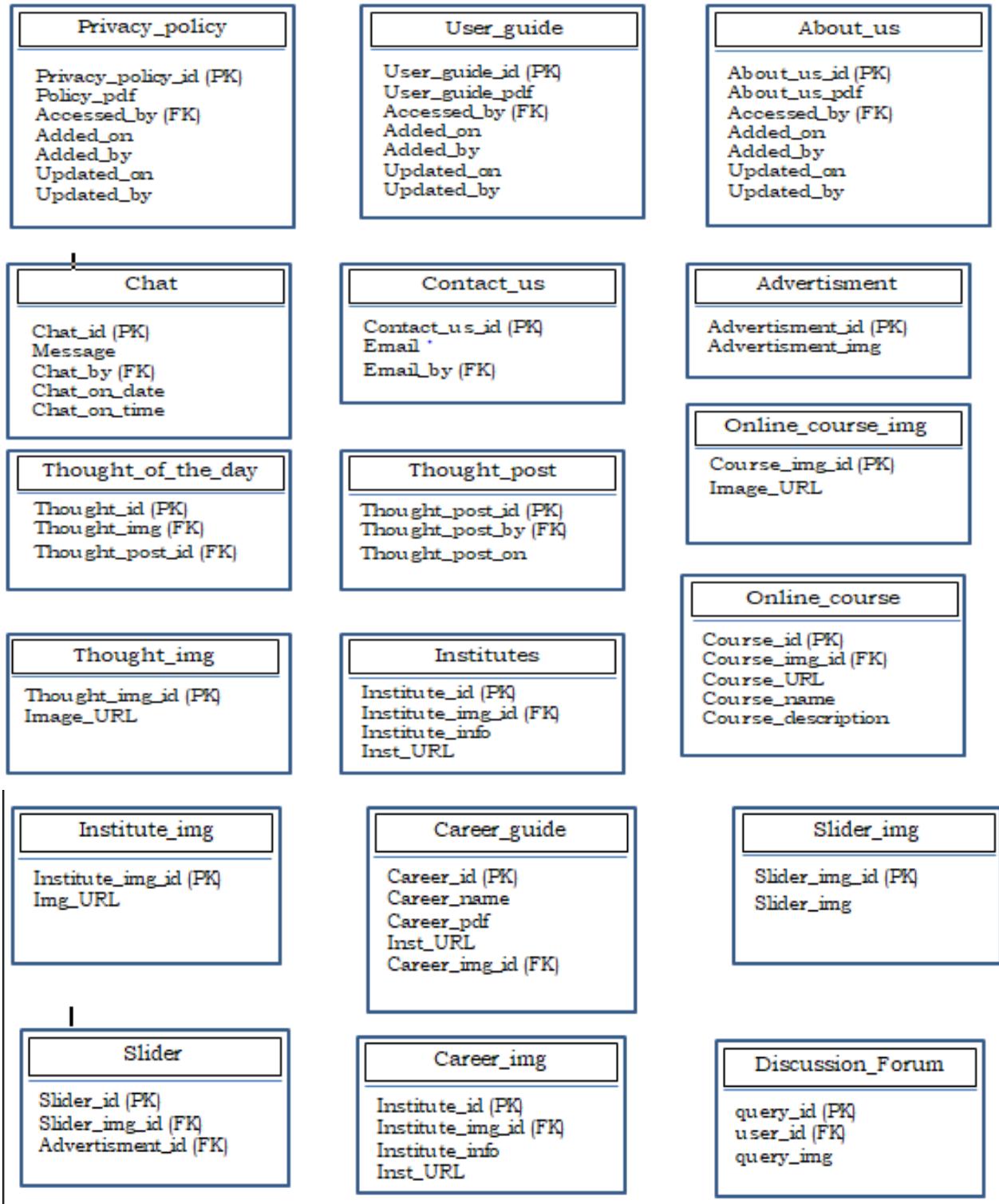
There are a few major reasons why Book-Exchange is written as an app.

The first reason is that as of 68% of people in the Pakistan uses a smart phone (android). Smart phones were also be chosen because of their technical abilities. They provide a built in camera that can be used to scan barcode of that book. Lastly, smartphones are good medium to collect data on, since most people do not share smartphones with others, whereas a computer might be shared within a household.

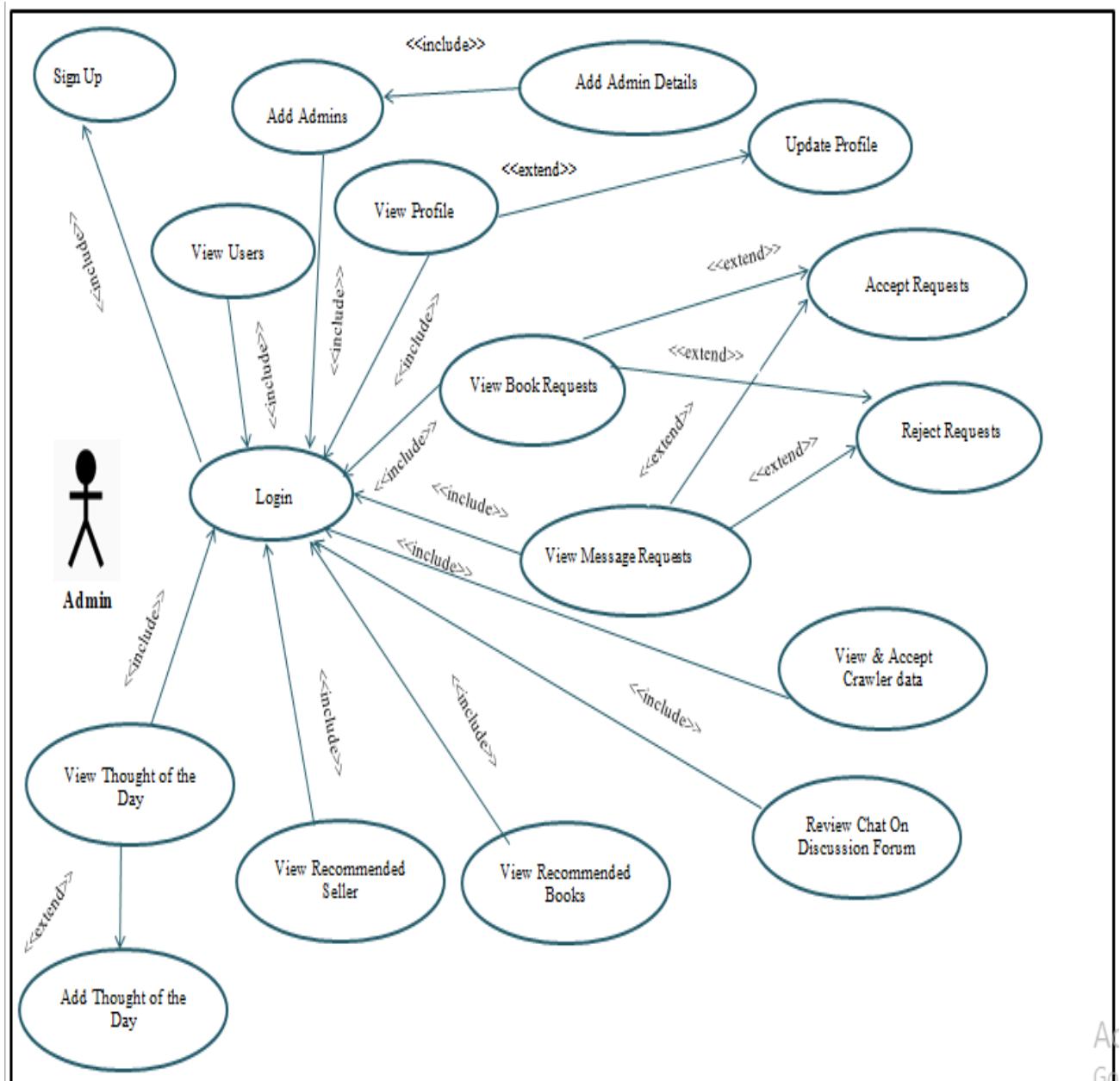


ERD



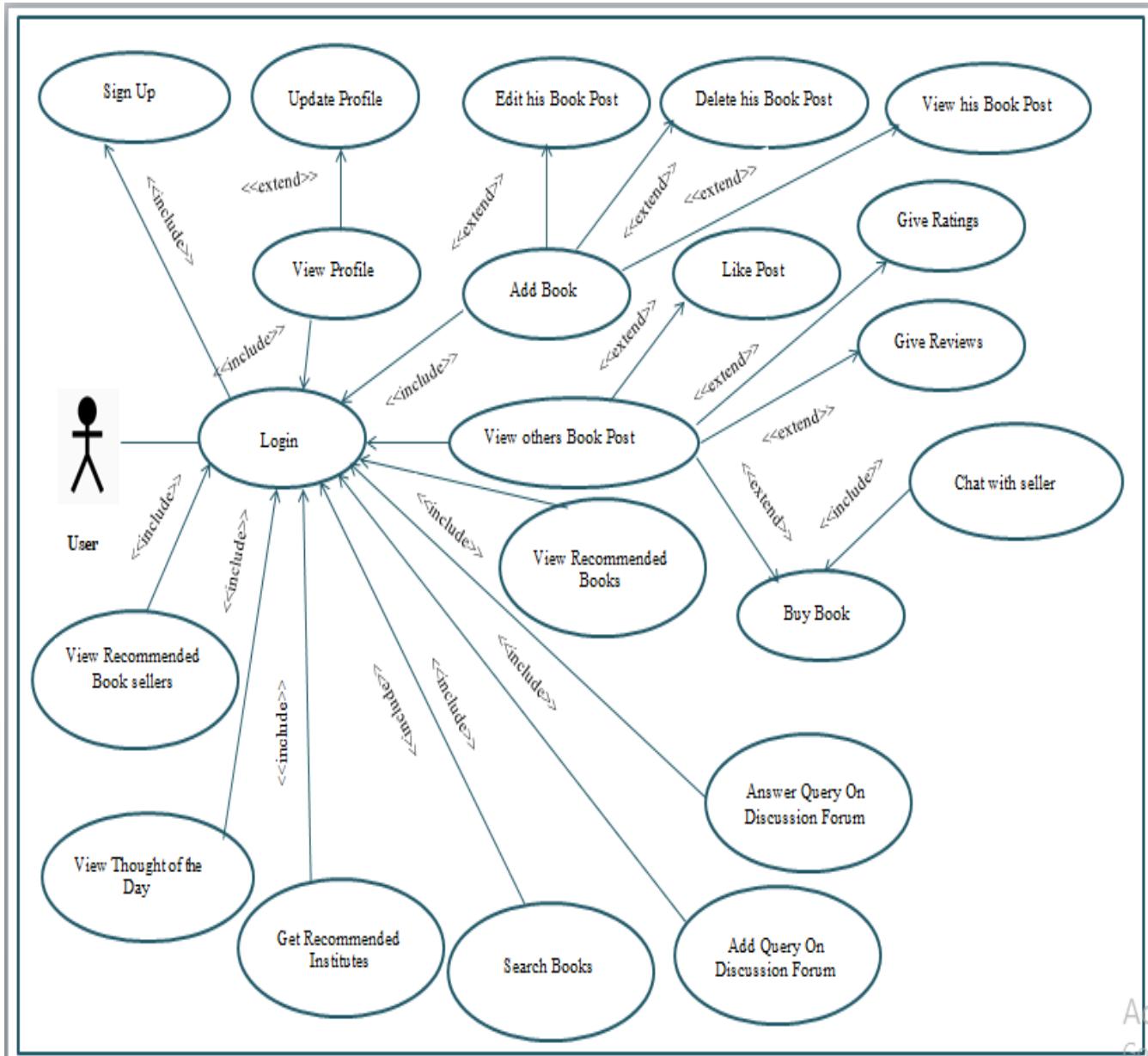


Use Cases (Admin Side)

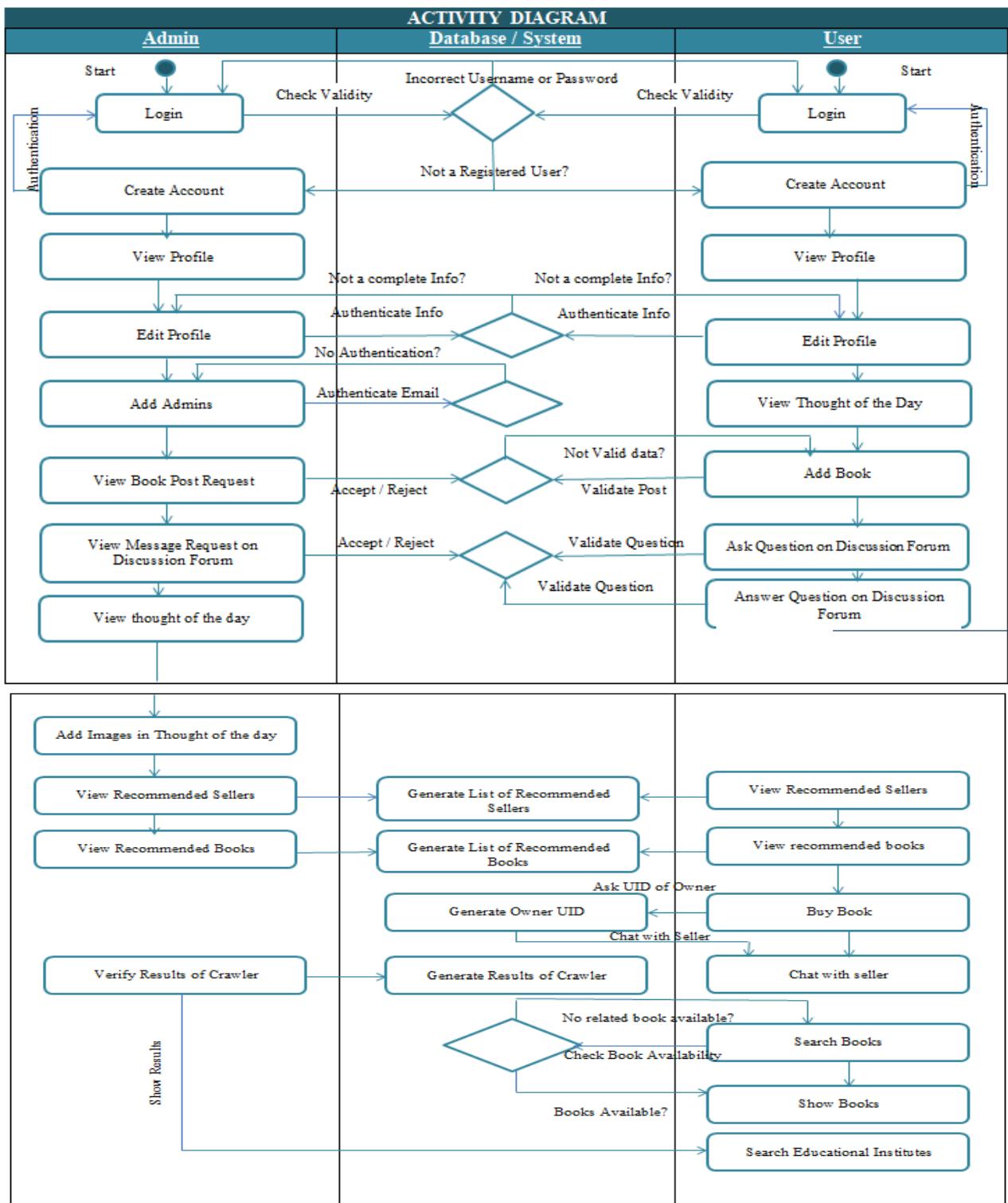


A
G

Use Cases (User Side)

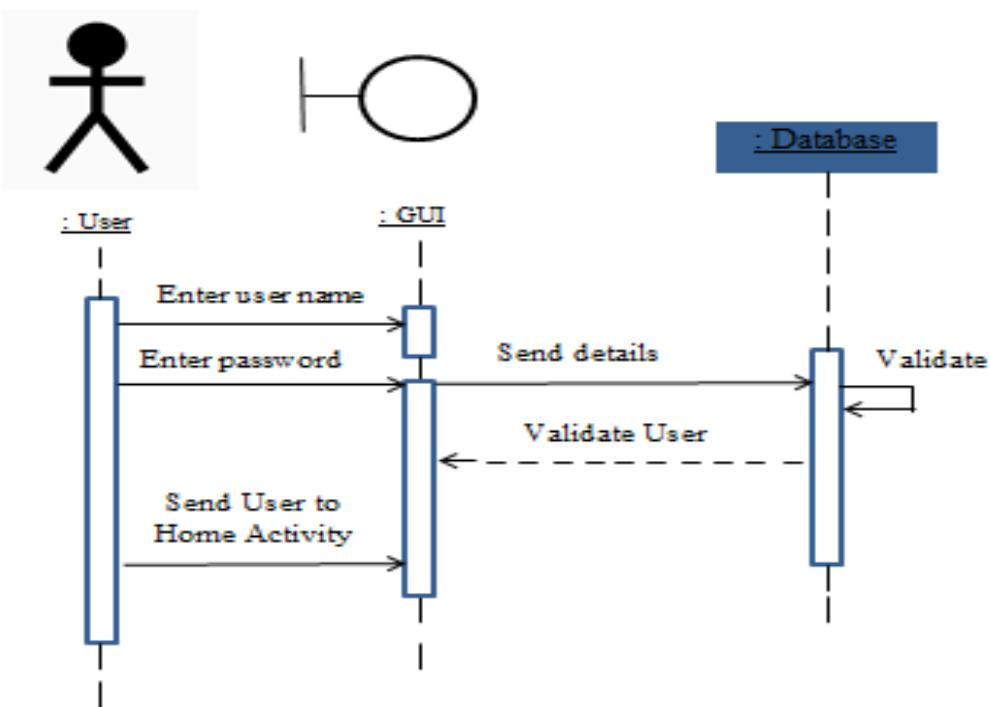


Activity Diagram (User and Admin Side)



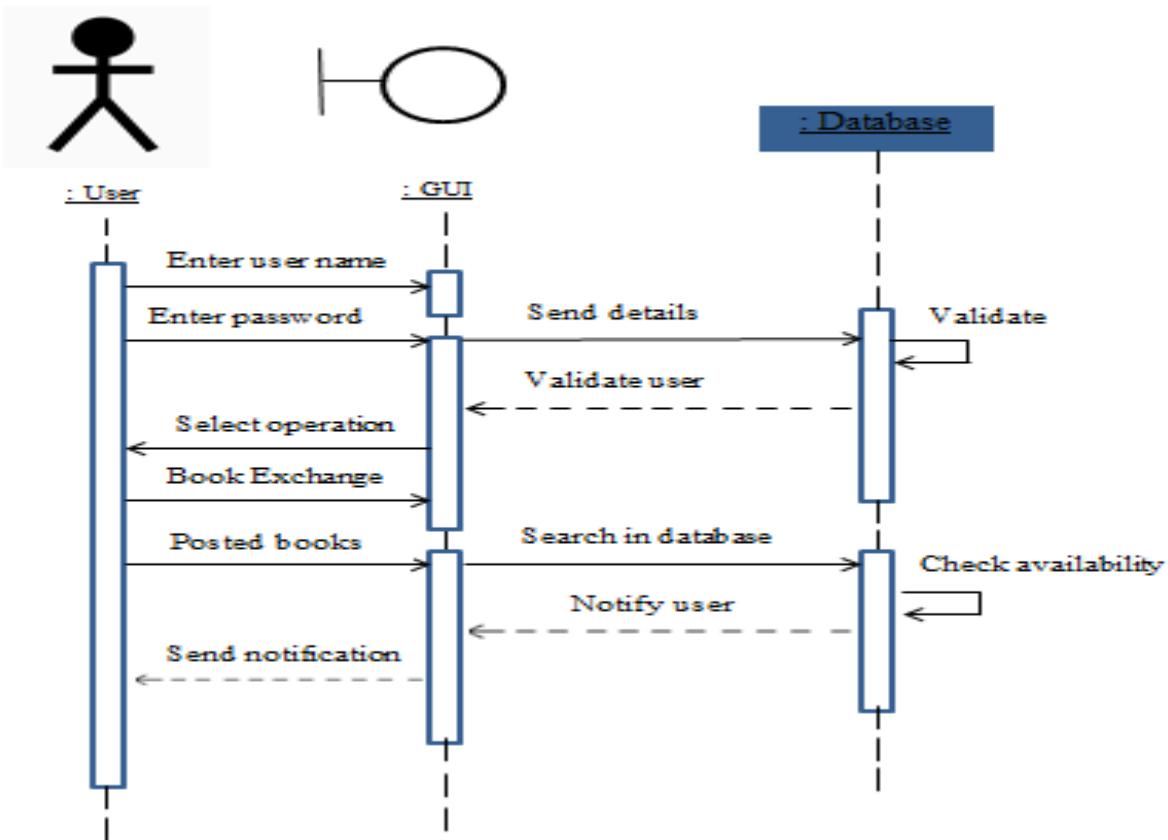
Sequence Diagrams (User Side)

For validating a user



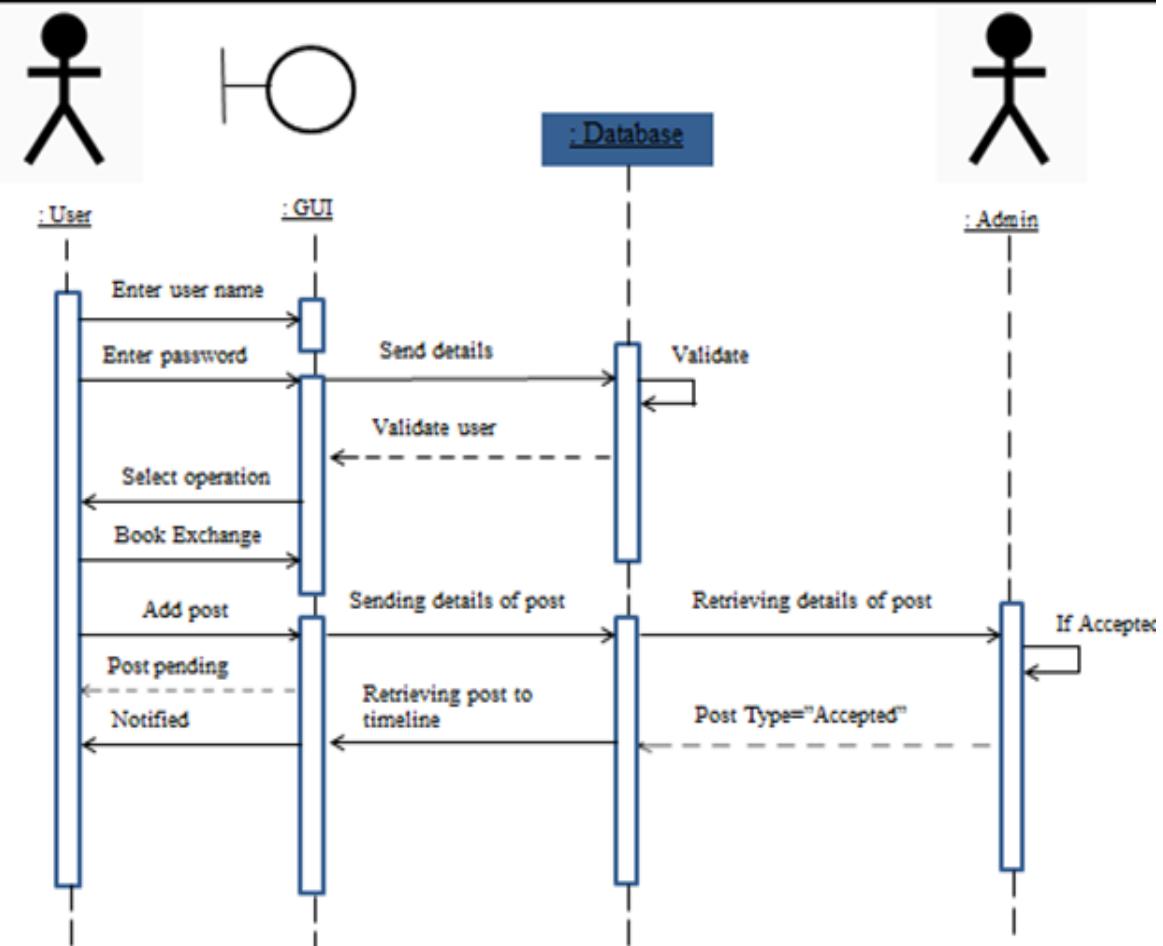
Sequence Diagrams (User Side)

For searching desired book as a user



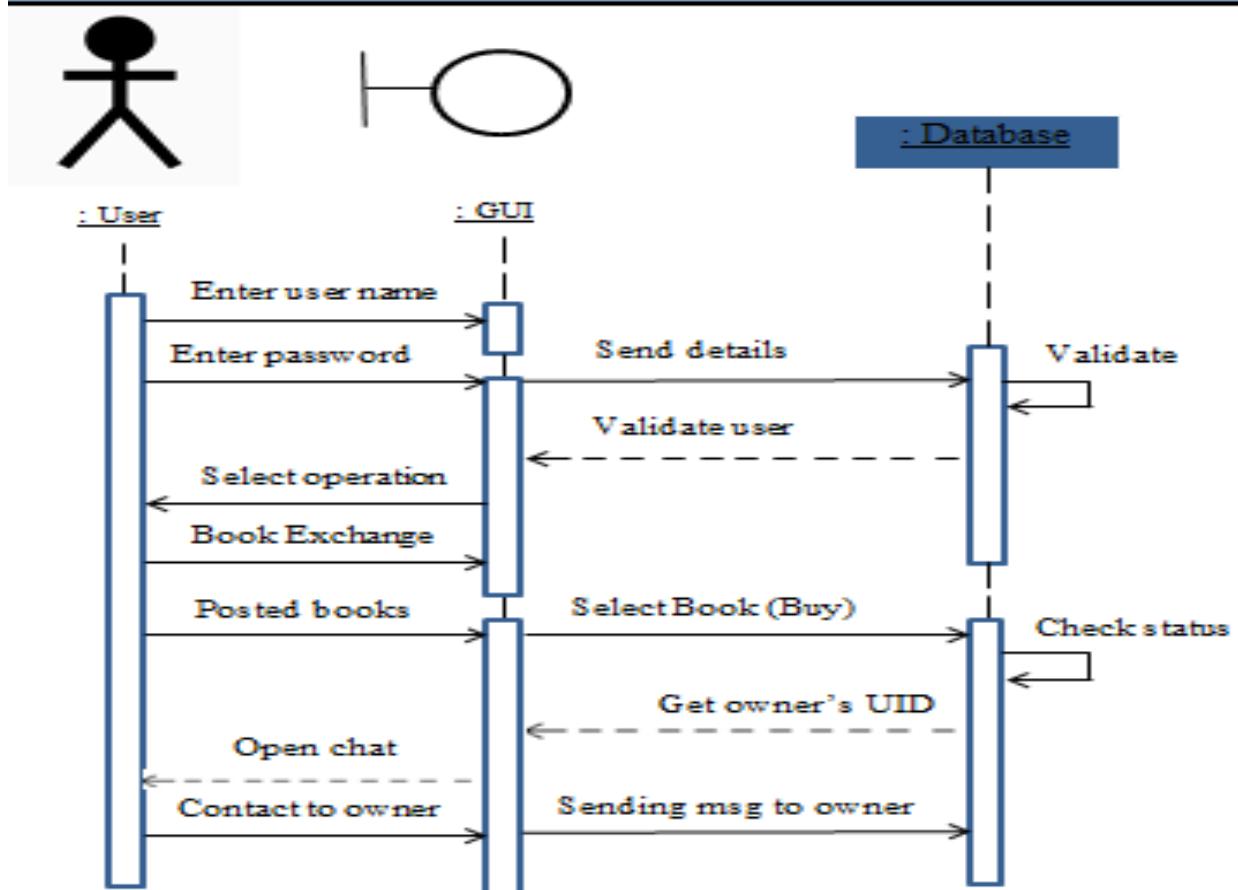
Sequence Diagrams (User Side)

For accessing book exchange as a seller



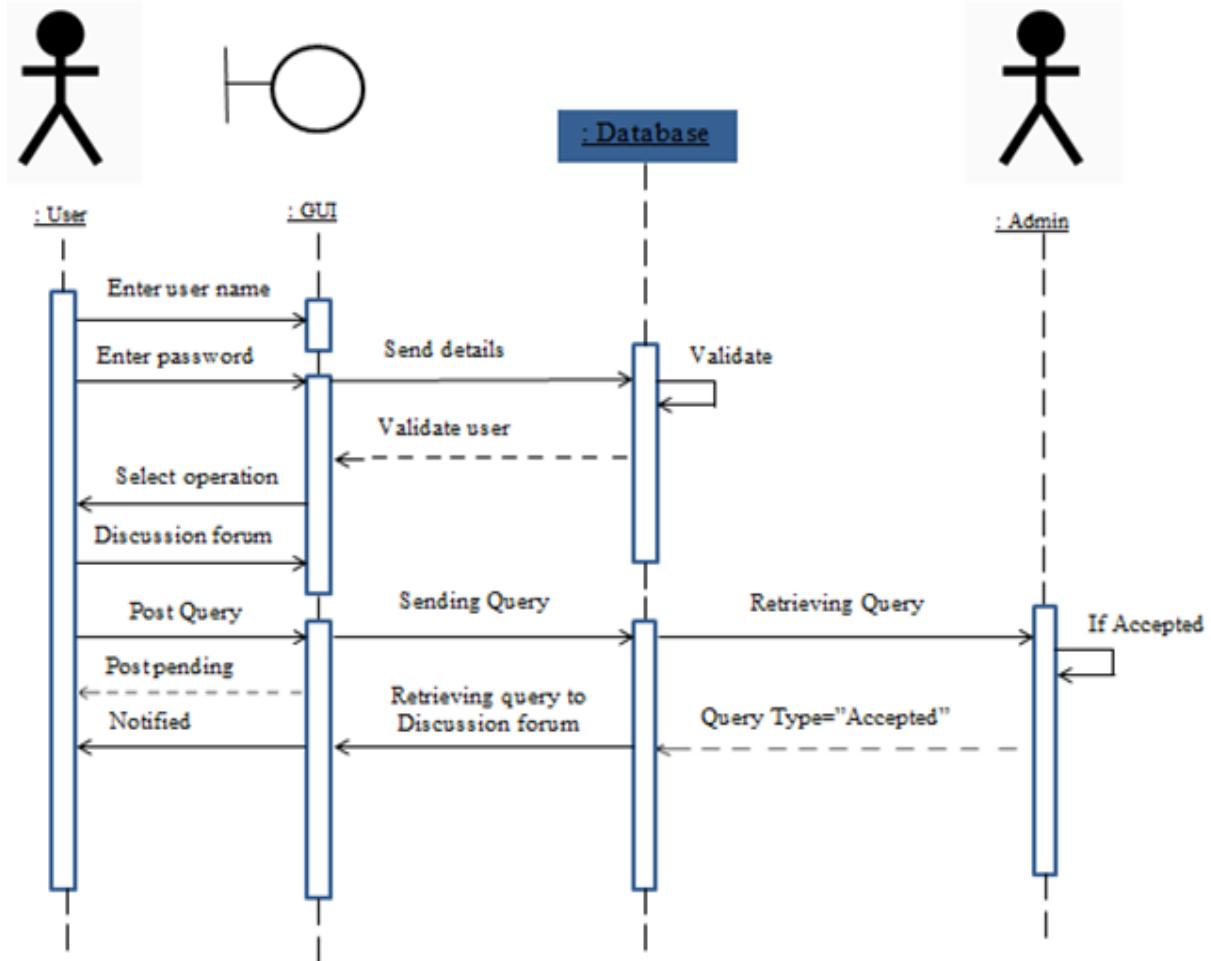
Sequence Diagrams (User Side)

For accessing book exchange as a buyer



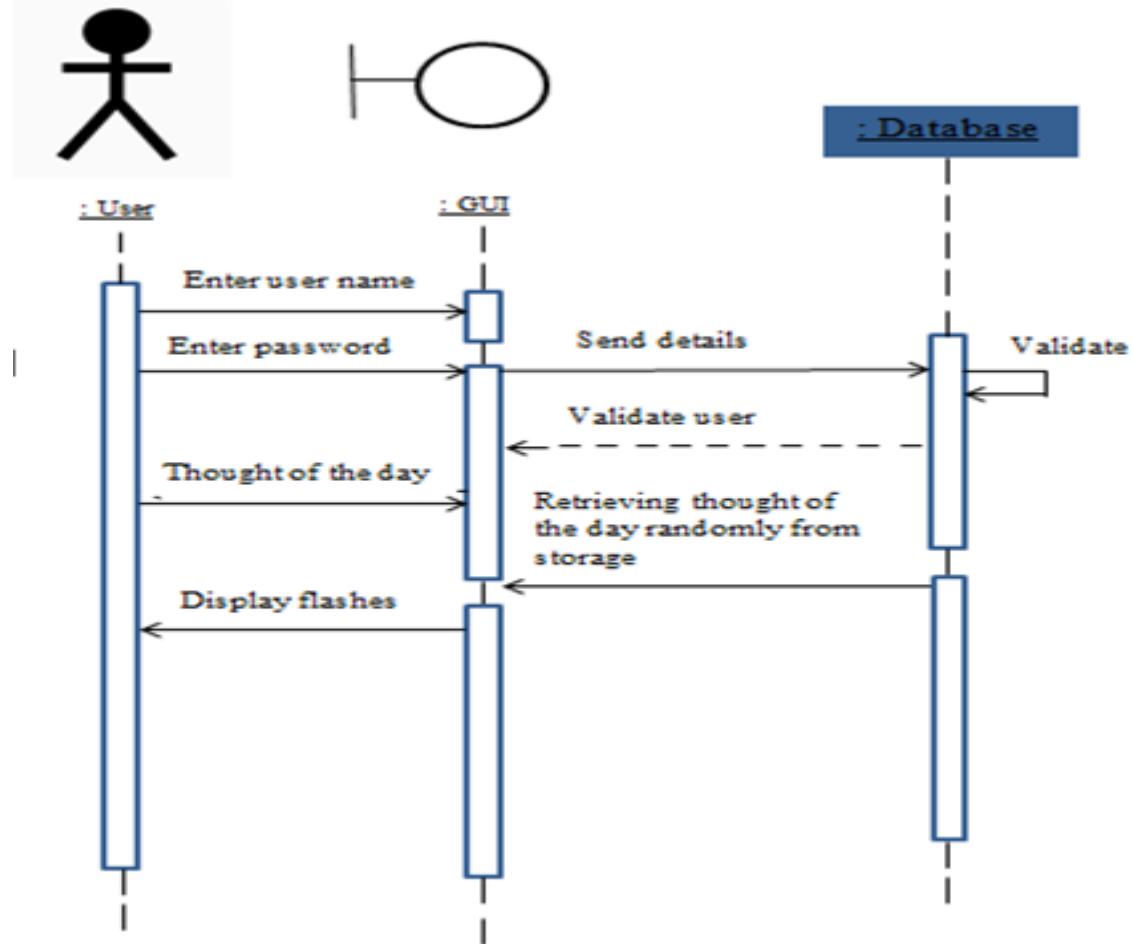
Sequence Diagrams (User Side)

For accessing Discussion Forum



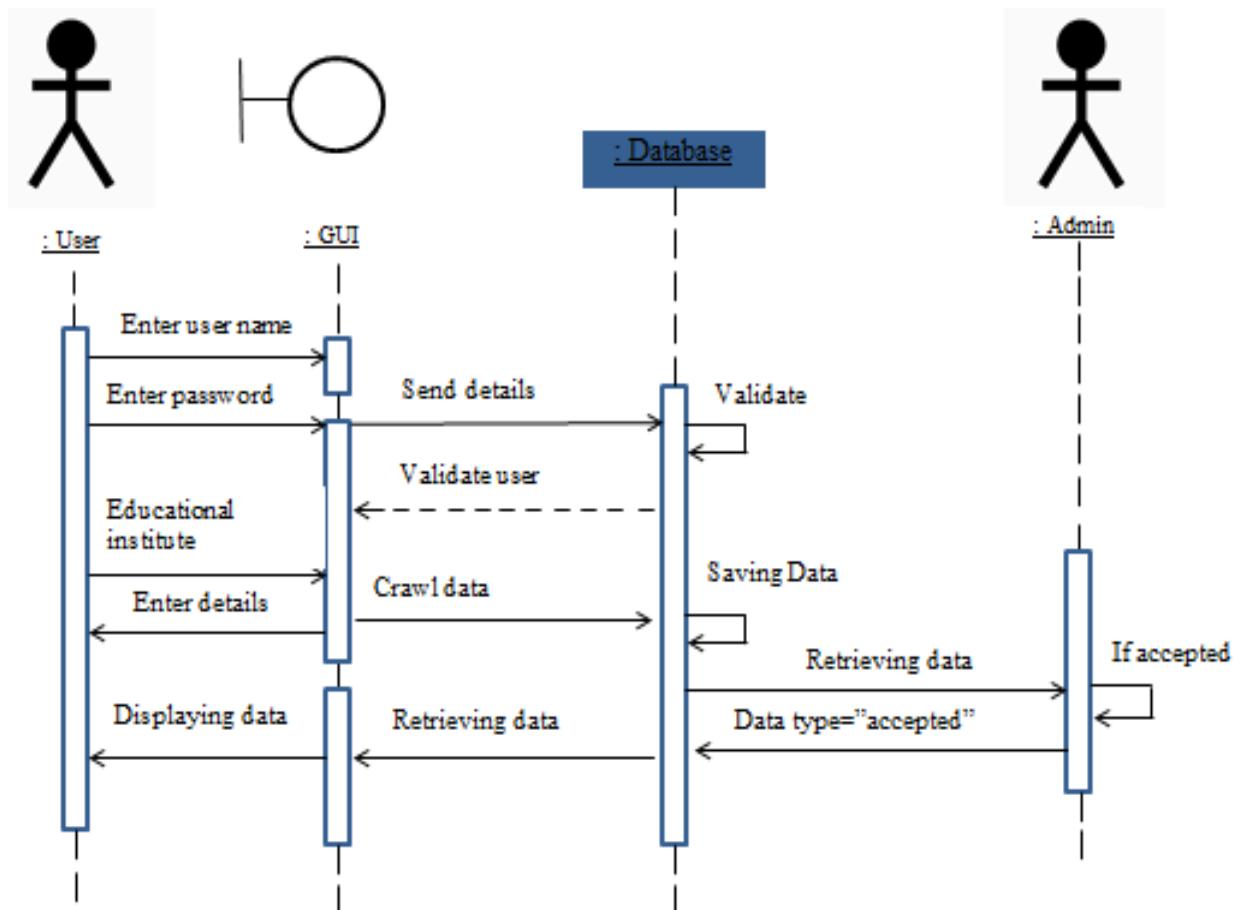
Sequence Diagrams (User Side)

For accessing Thought of the day



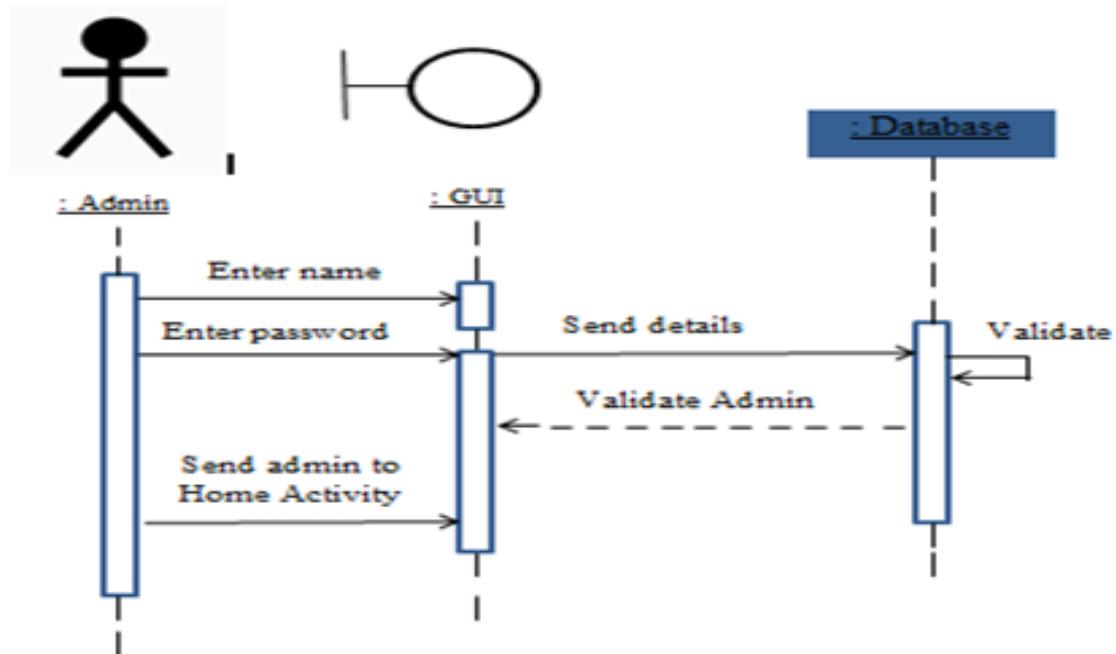
Sequence Diagrams (User Side)

For accessing Educational Institute



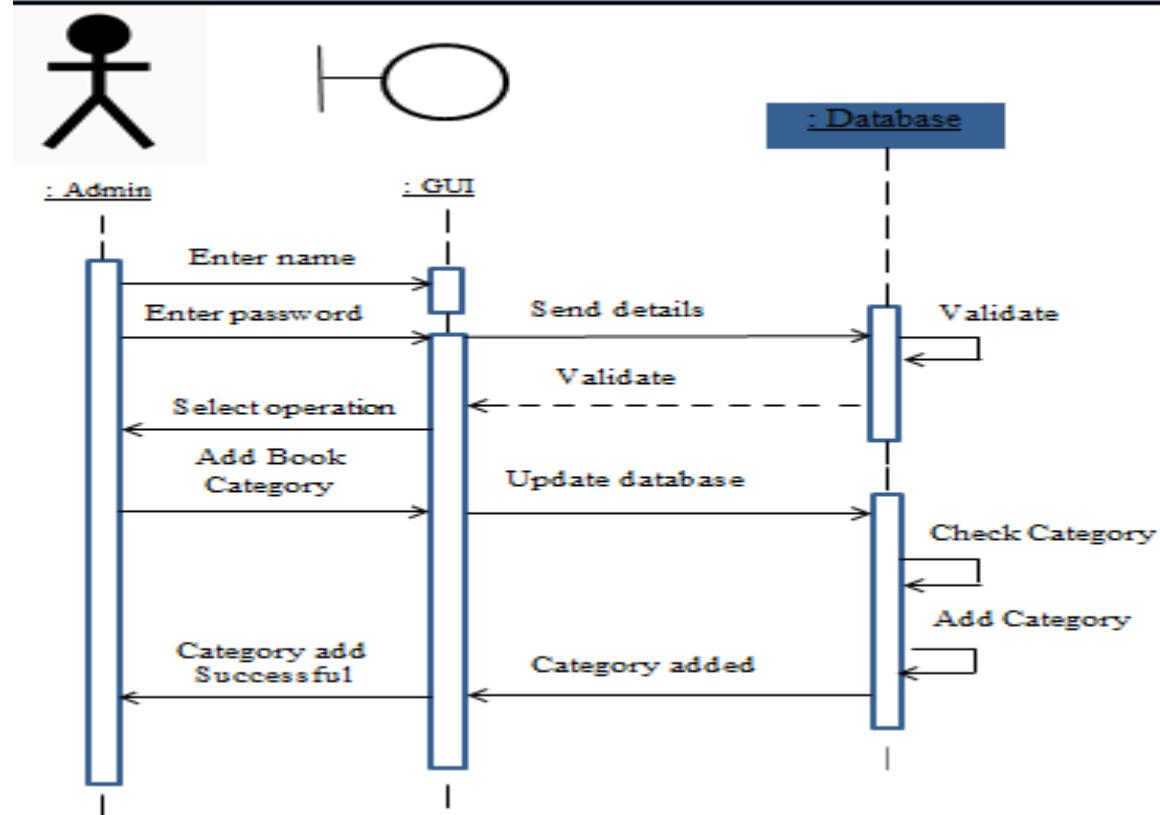
Sequence Diagrams (Admin Side)

For validating Admin



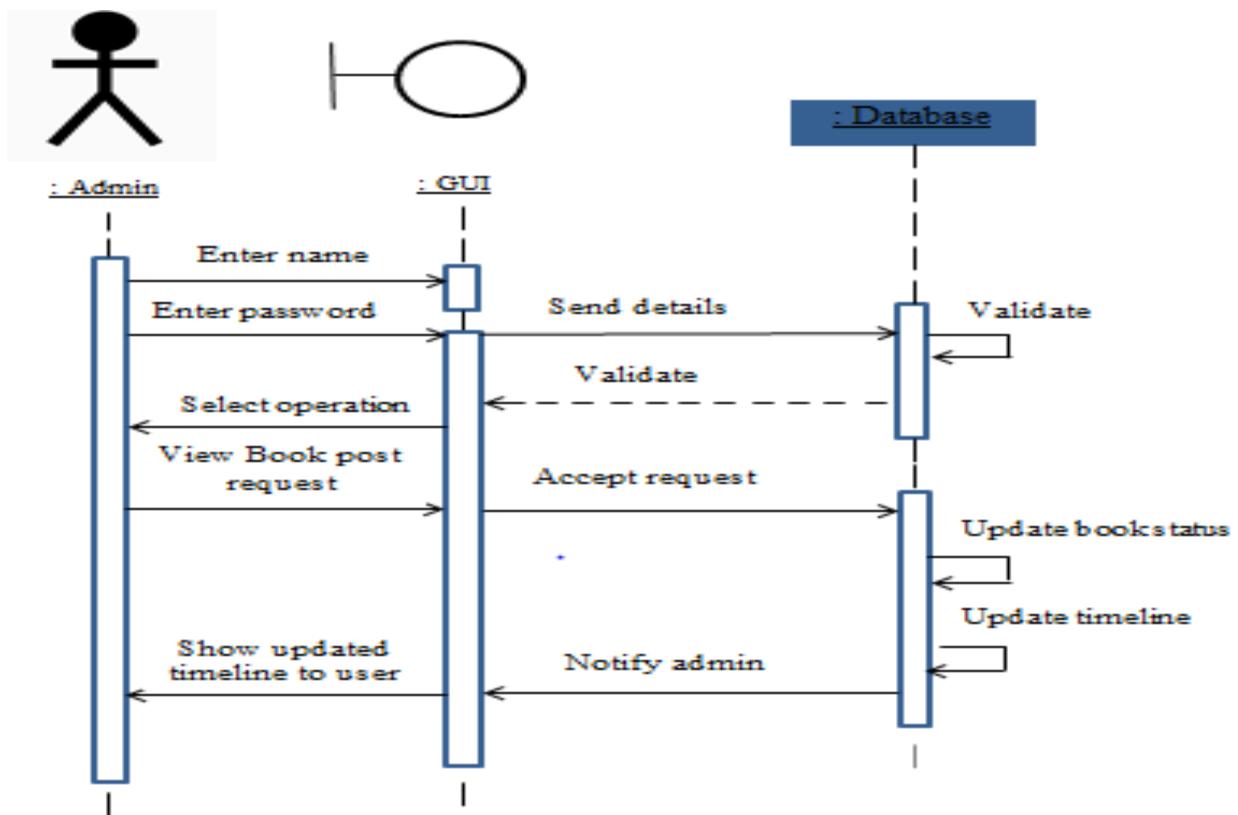
Sequence Diagrams (Admin Side)

For Admin Add Book Category



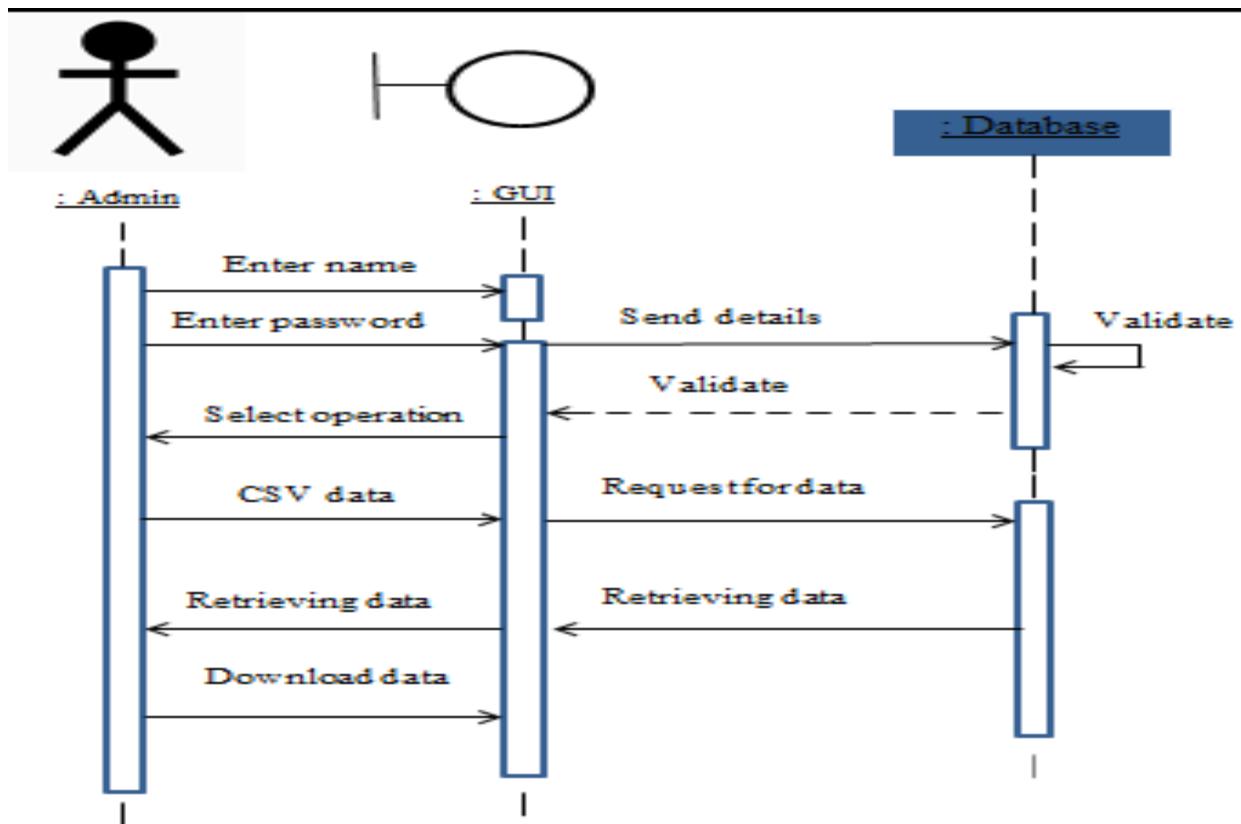
Sequence Diagrams (Admin Side)

For Admin Accepting Book Post Request



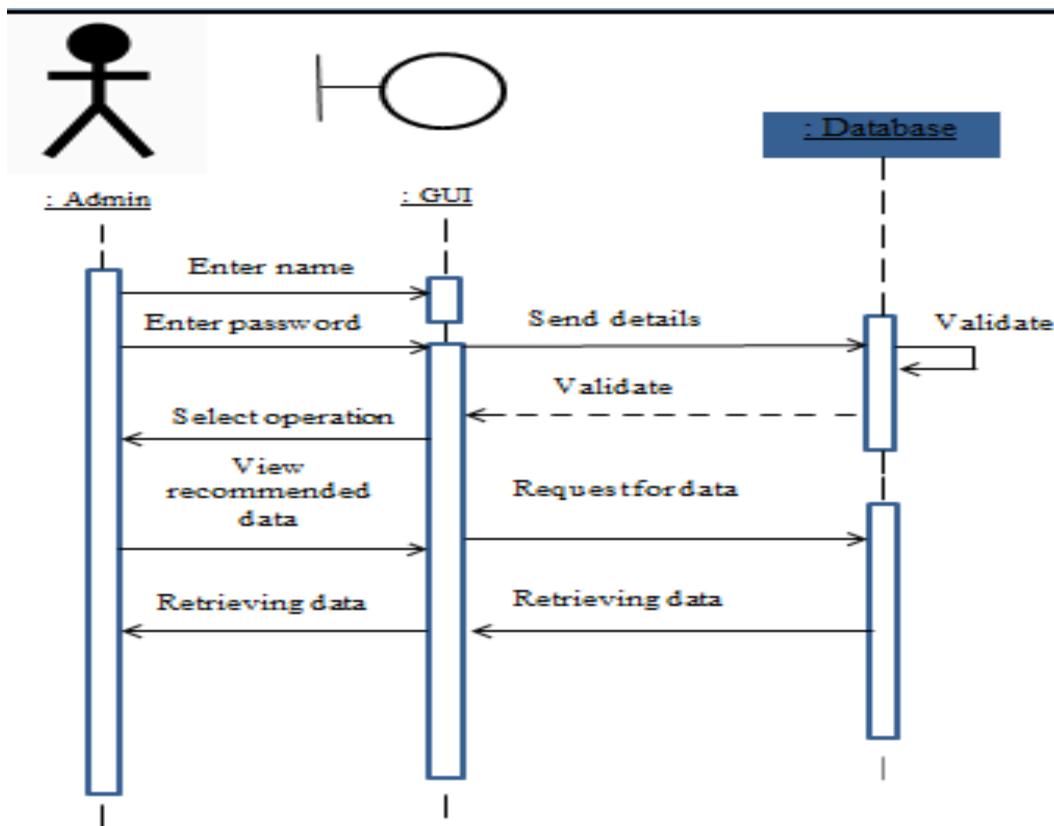
Sequence Diagrams (Admin Side)

For Admin Download CSV



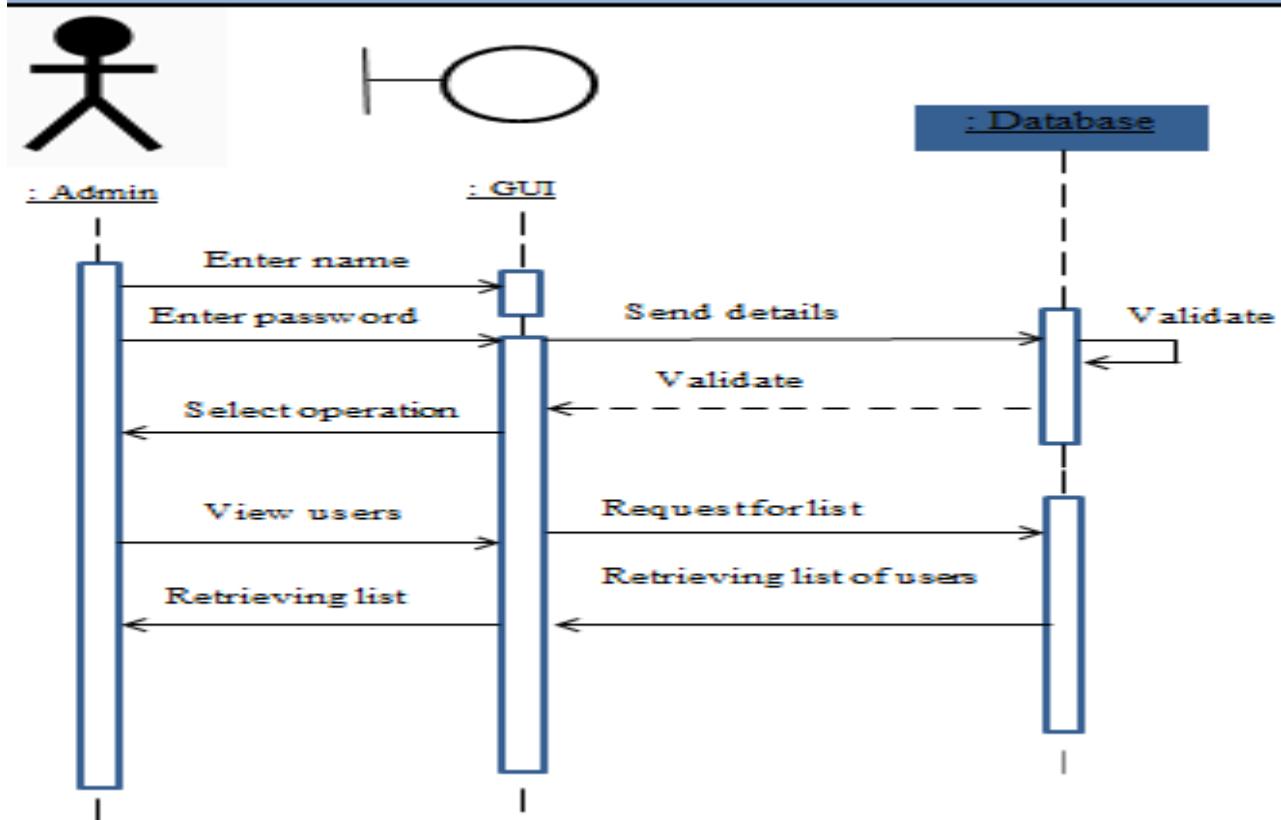
Sequence Diagrams (Admin Side)

For Admin View Recommended data



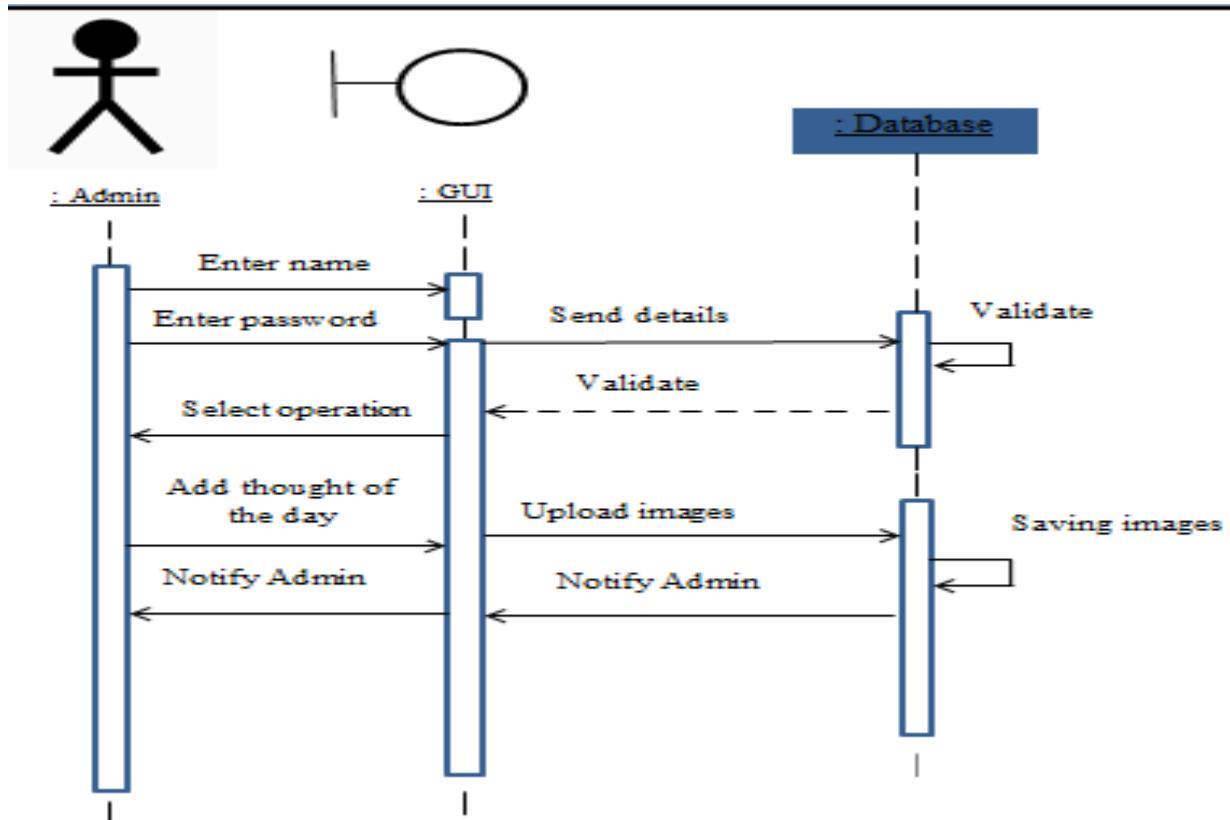
Sequence Diagrams (Admin Side)

For Admin View List of users



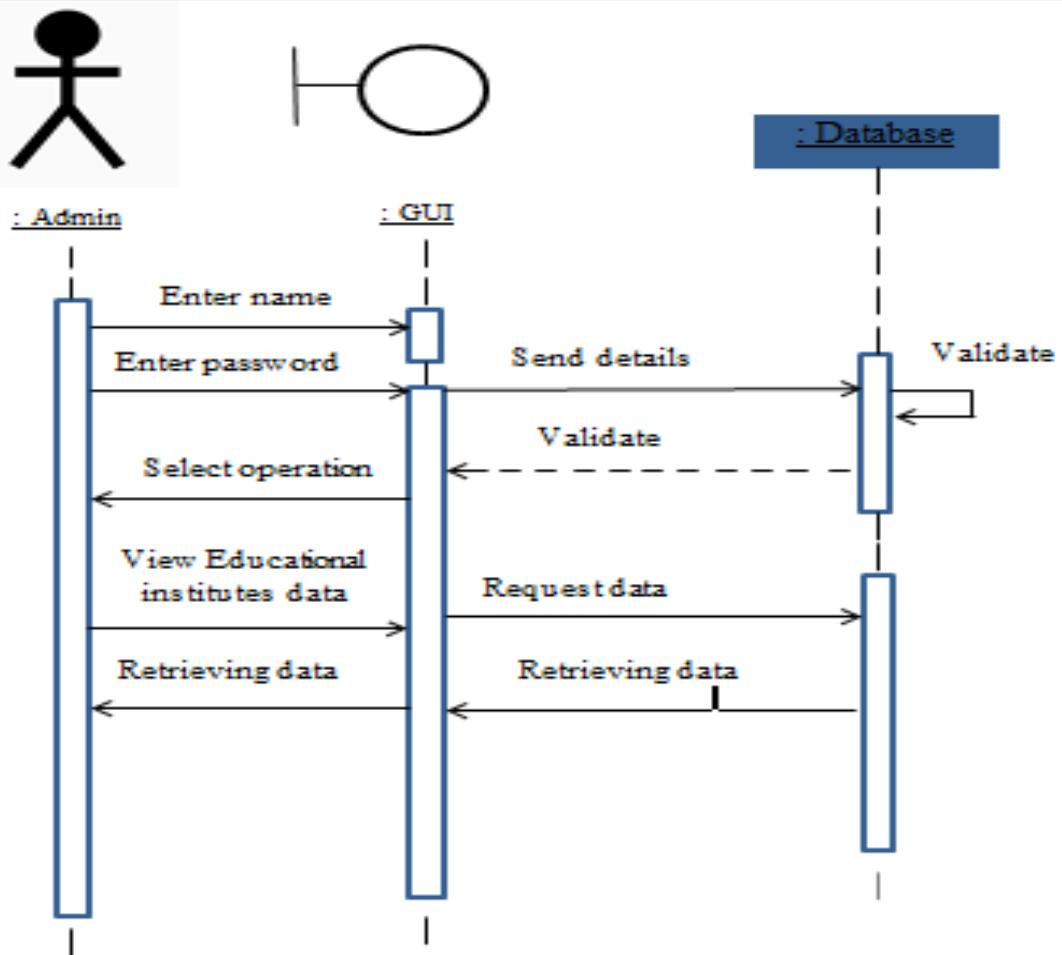
Sequence Diagrams (Admin Side)

For Admin Uploading Thought of the day Images

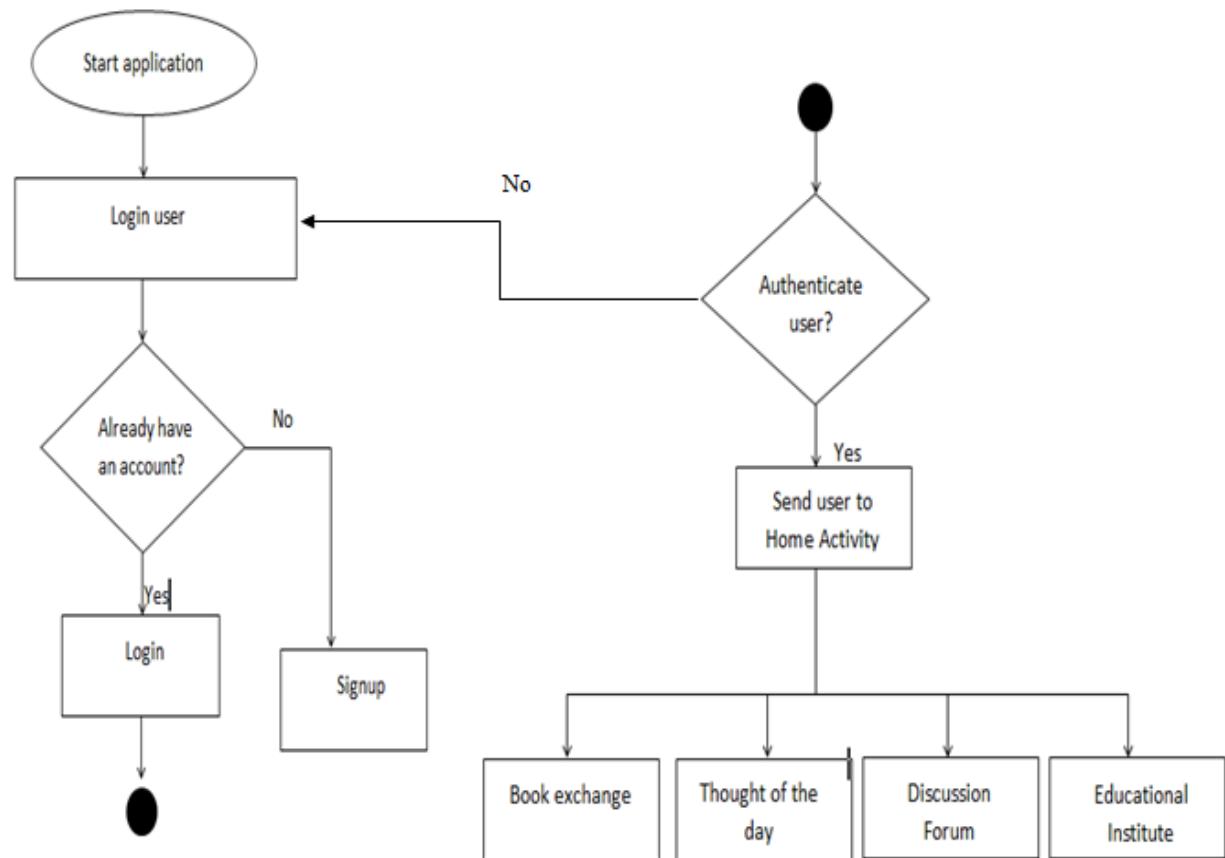


Sequence Diagrams (Admin Side)

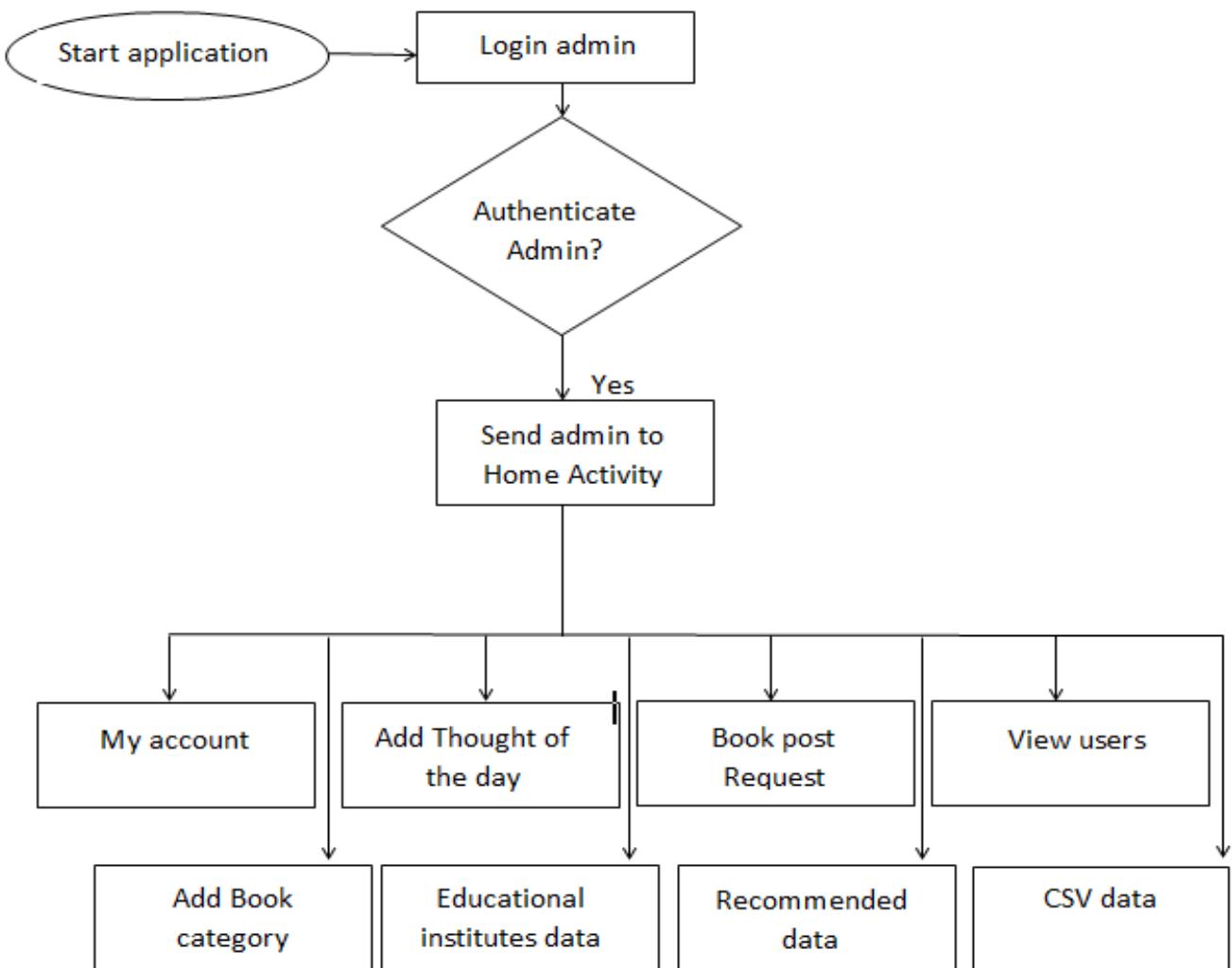
For Admin Viewing Educational Institutes Data



Flow Diagrams (User Side)

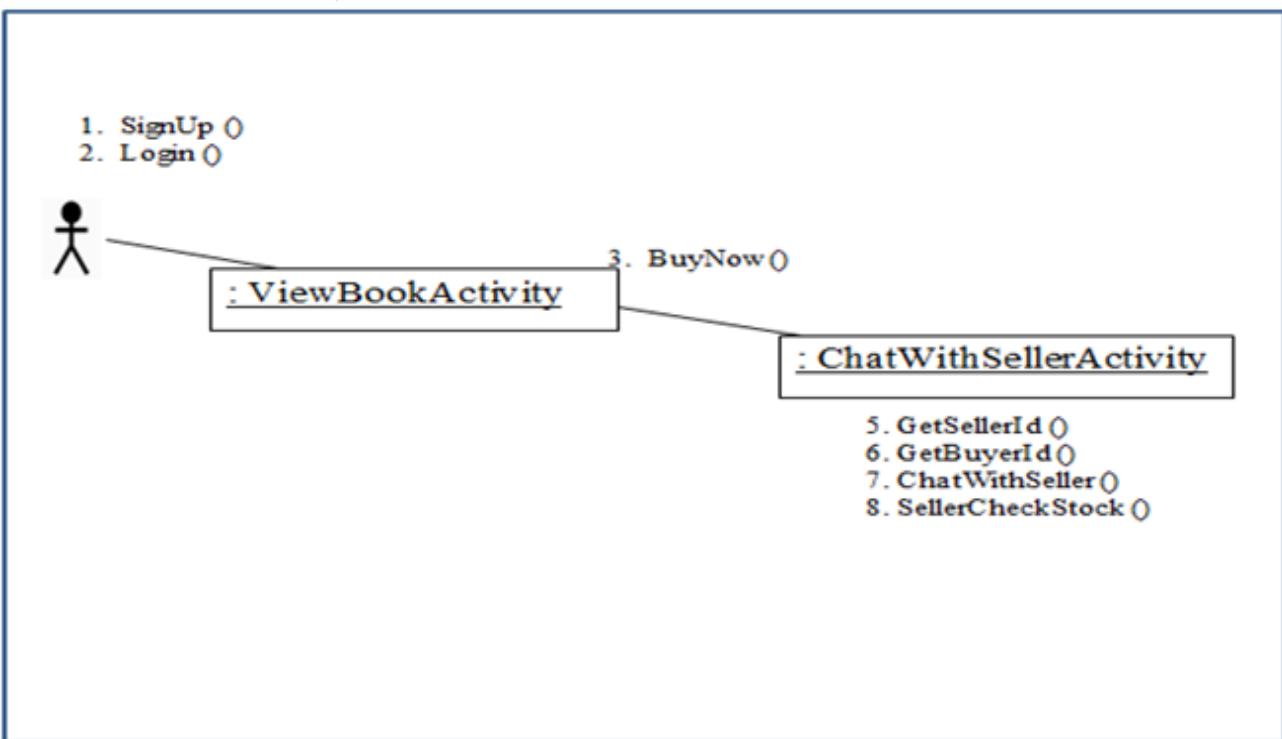


Flow Diagrams (Admin Side)



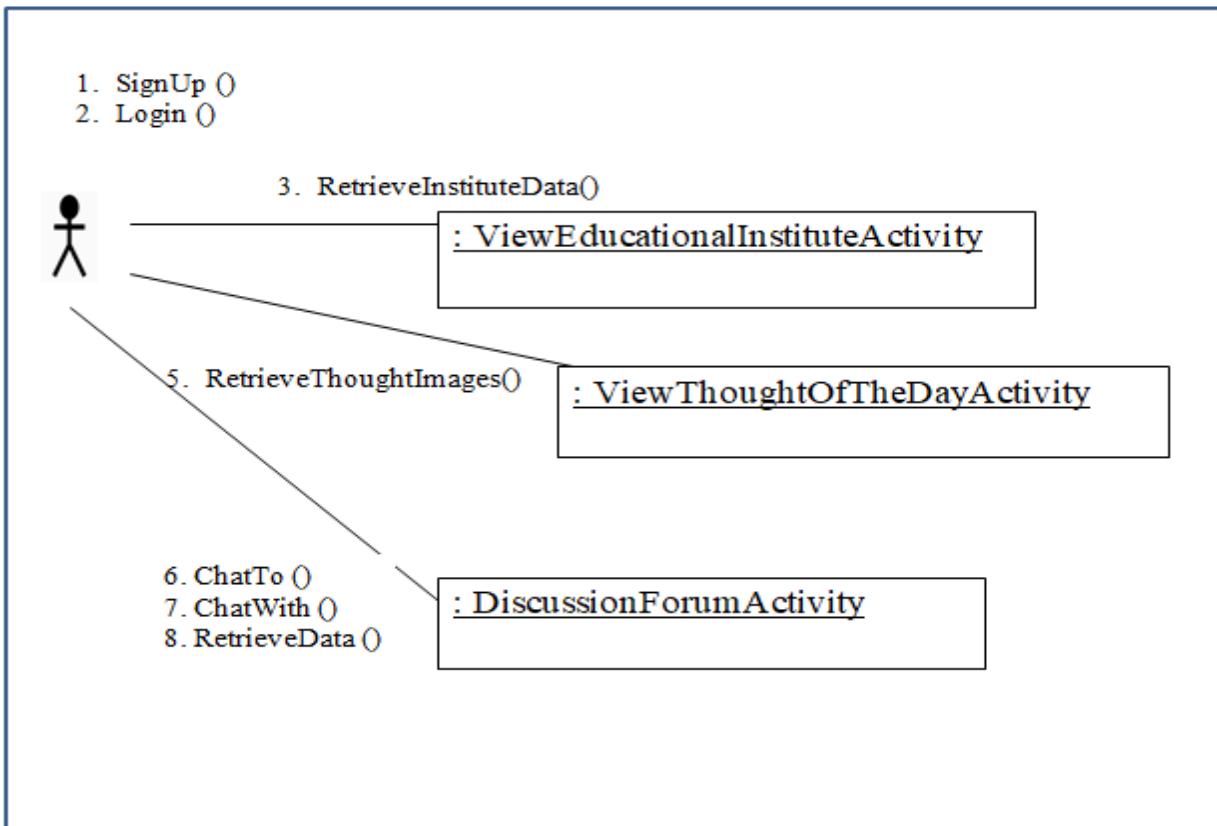
Collaboration Diagrams (User Side)

Scenario when user buys a book

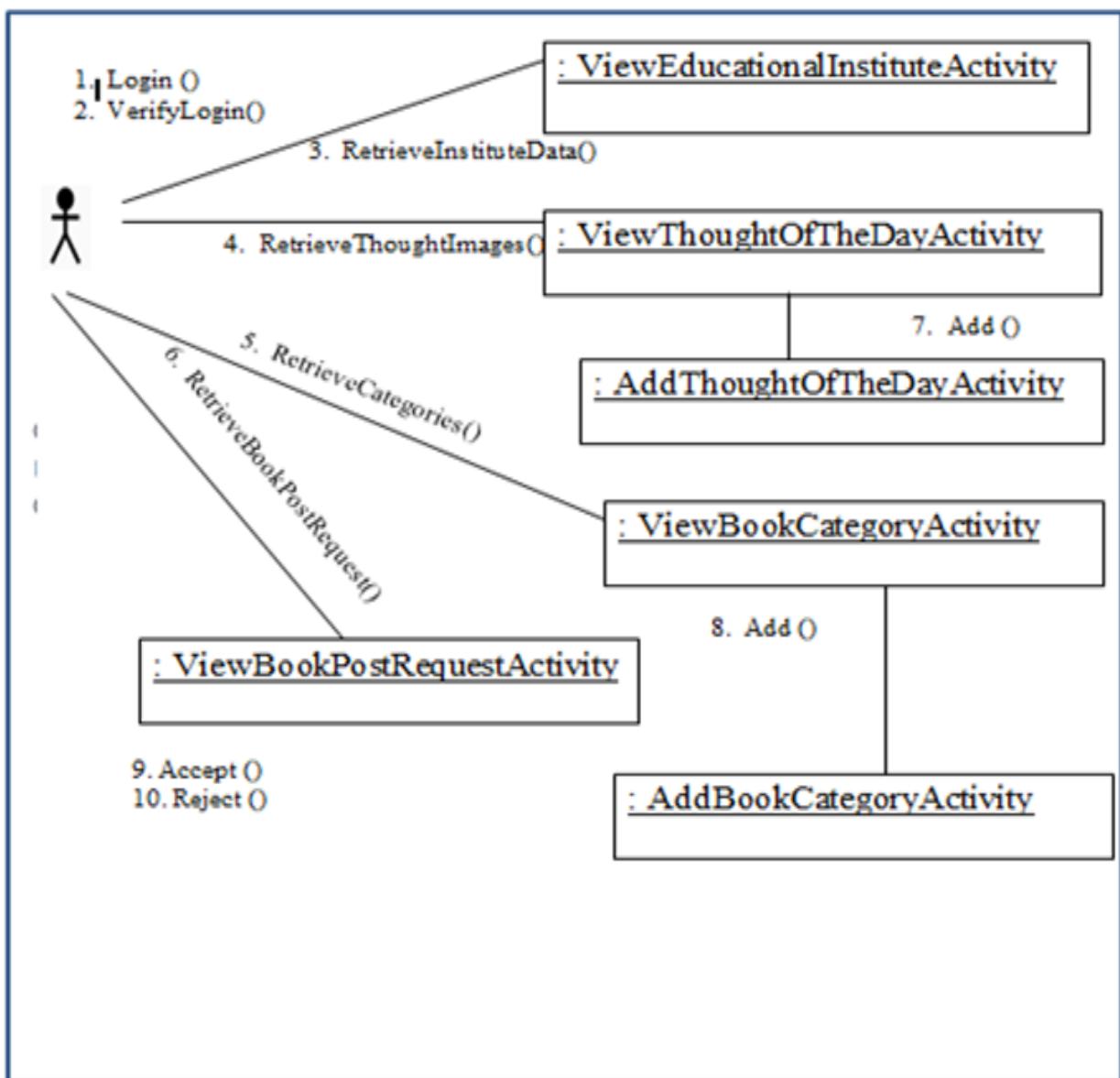


Collaboration Diagrams (User Side)

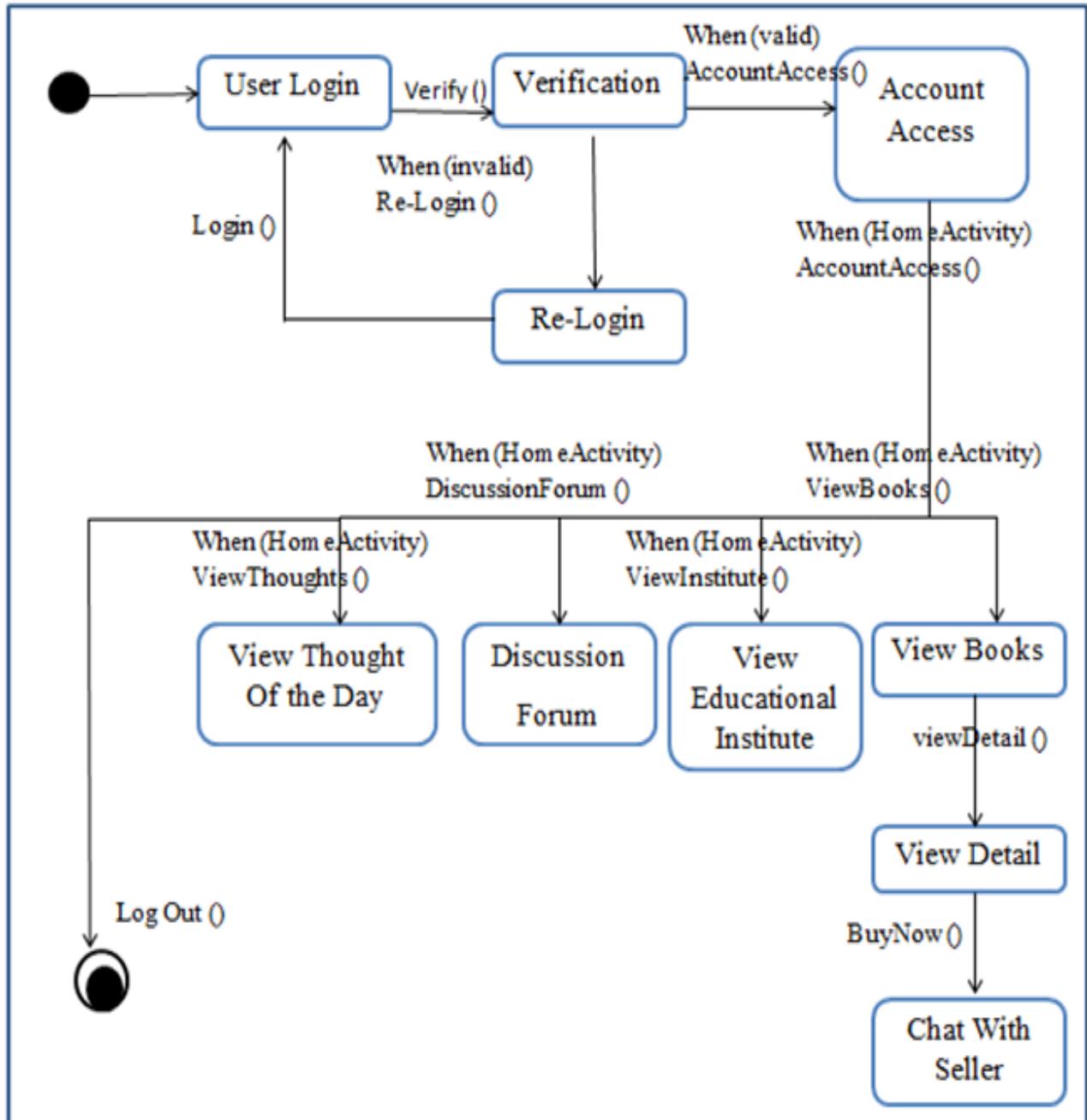
Scenario when user Login



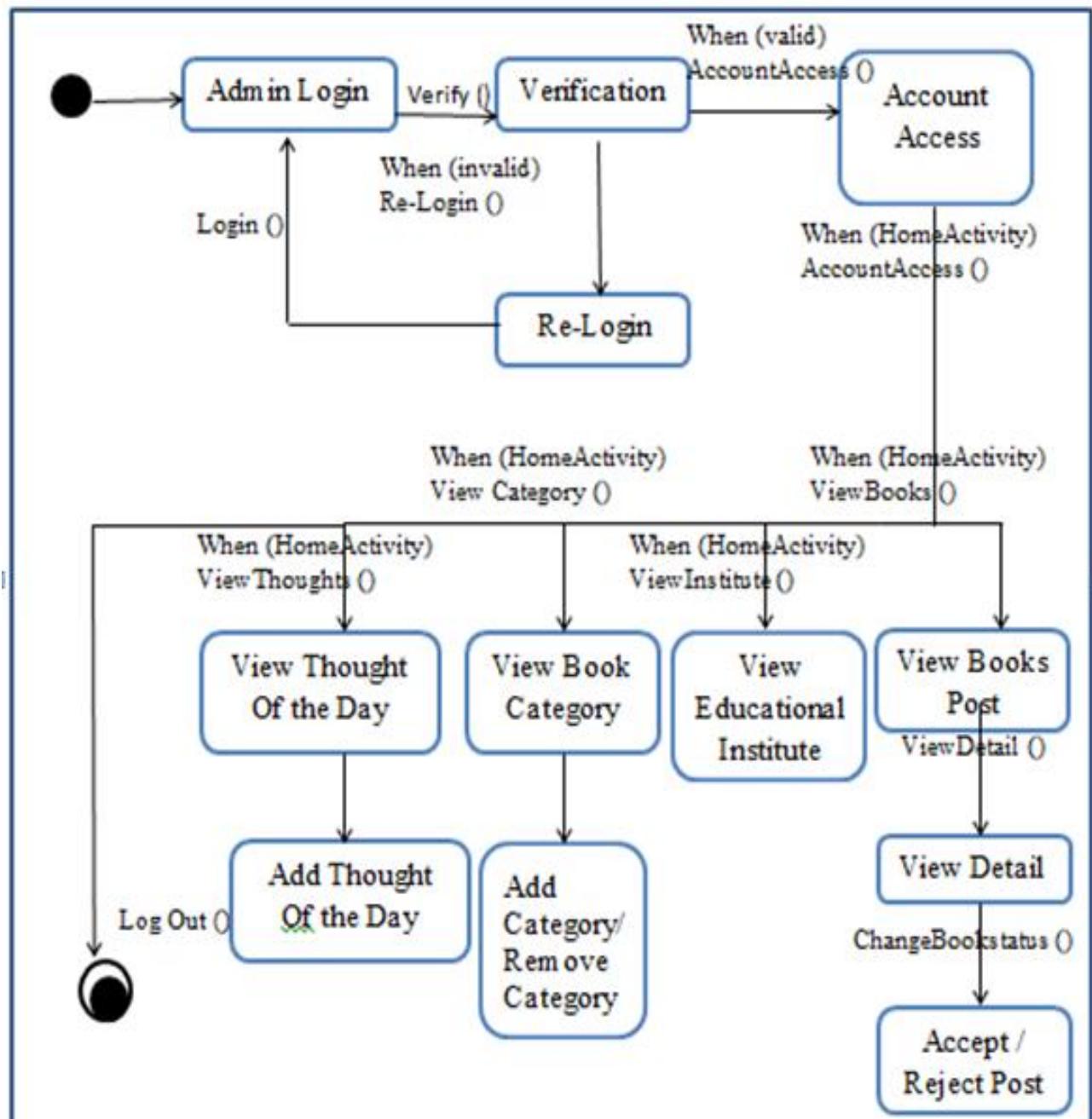
Collaboration Diagrams (Admin Side)



State Chart Diagrams (User Side)



State Chart Diagrams (Admin Side)

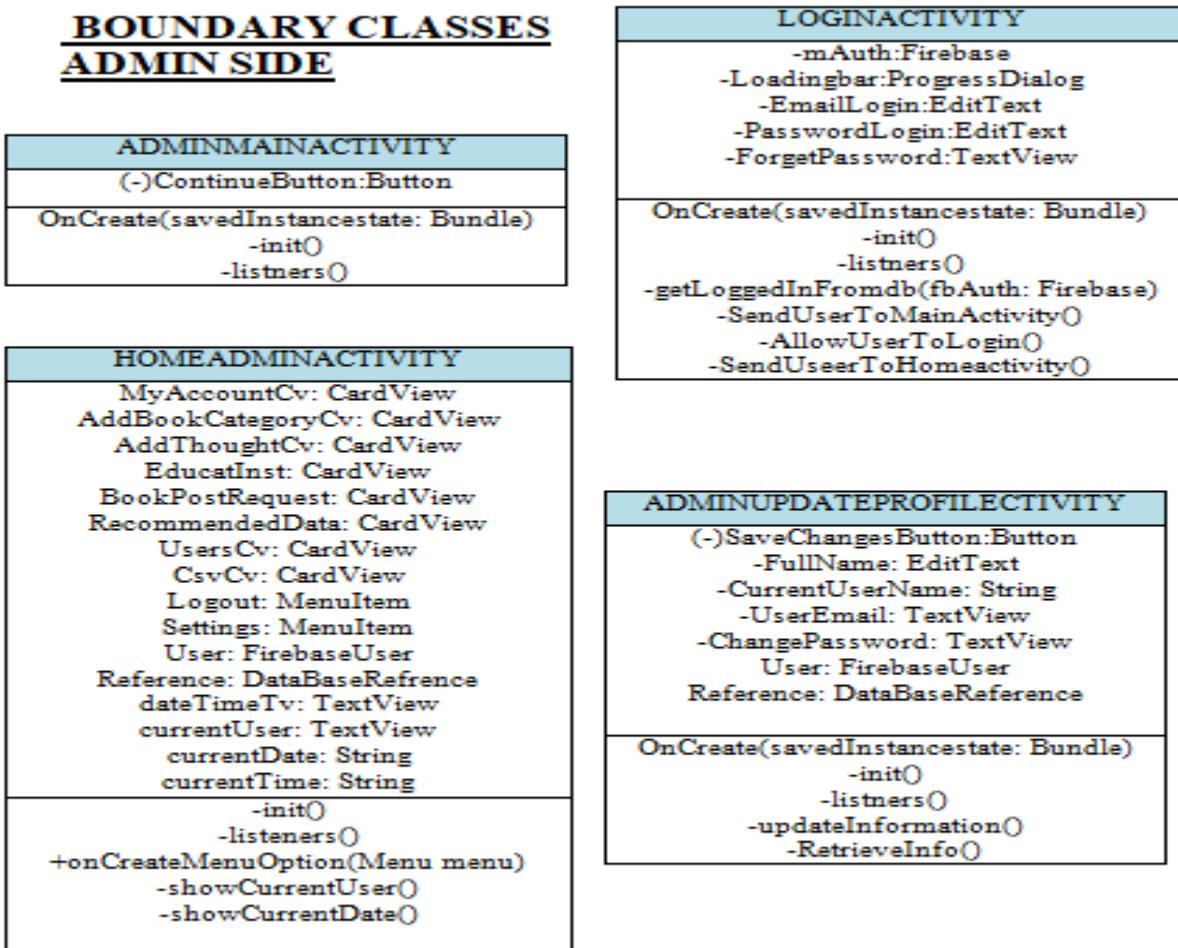


CLASS DIAGRAM ADMIN SIDE

CLASS DIAGRAM ADMIN SIDE

There are three types of classes.

- Boundary Classes (i.e. Activity Classes)
- Entity Classes (i.e. Model Classes)
- Controller Classes (i.e. Adapter Classes)



ADDBOOKCATEGORYACTIVITY

```
-addNewCatEt: EditText
-AddNewCatBtn: Button
loadingBar: ProgressDialog
calendar: Calender
currentDate: String
currentTime: String
addedOn: long
currentUser: Firebase
reference: DatabaseReference
addedBy: String
productRandomKey: String
OnCreate(savedInstanceState: Bundle)
    Init()
    Listners()
    AddBookCategory()
```

ADMINEDUCATIONALINSTITUTEACTIVITY

```
Toolbar: ToolBar
Viewpager: ViewPager
tabLayout: TabLayout
myEtAdapter: TabAccessordadapter
OnCreate(savedInstanceState: Bundle)
```

ADDTHOUGHTIMAGESACTIVITY

```
-addNewImage: ImageView
-UploadBtn: Button
loadingBar: ProgressDialog
calendar: Calendar
currentDate: String
currentTime: String
addedOn: long
currentUser: Firebase
reference: DatabaseReference
addedBy: String
productRandomKey: String
imageUri: Uri
saveCurrentDate: String
saveCurrentTime: String
OnCreate(savedInstanceState: Bundle)
    Init()
    Listners()
    validateImageInfo()
    StoreImageToDb()
    saveImage()
```

BOOKPOSTREQUESTACTIVITY

```
postRecyclarView: Recycler View
Viewpost: Button
Adapter: BookPostAdapter
OnCreate(savedInstanceState: Bundle)
    Init()
    Listners()
    onStart()
    onStop()
```

FORGETPASSWORDACTIVITY

```
Email: EditText
ResetBtn: Button
mAuth: Firebase
loadingbar: ProgressDialog
Init()
Listners()
ResetPassword()
```

SHOWBOOKCATEGORIEACTIVITY

```
Recycle: RecyclerView
Addcat: Button
Adapter: CategoryAdapter
Init()
Listners()
onStart()
onStop()
```

BOOKSTATUSACTIVITY

```
Toolbar: ToolBar
Viewpager: ViewPager
tabLayout: TabLayout
myEtAdapter: TabAccessordadapter
OnCreate(savedInstanceState: Bundle)
```

THOUGHTIMAGESACTIVITY

```
AddNewThought: Button
ShowImages: RecyclerView
OnCreate(savedInstanceState: Bundle)
    Init()
    Listners()
```

VIEWFULLBOOKPOSTACTIVITY

```
BookName: TextView
AuthorName: TextView
BookCondition: TextView
BookCategory: TextView
BookPrice: TextView
BookDescript: TextView
BookCondition: TextView
User: FirebaseUser
Reference: DatabaseReference
Storage: DataBaseStorageRef
ProfileImage: CircularImage
OnCreate(savedInstanceState: Bundle)
```

BOOKSTATUSACTIVITY

```
Init()
Listners()
RetrieveData()
```

VIEWSUSERSACTIVITY

```
recyclarView: RecyclerView
adapter: UserAdapter
OnCreate(savedInstanceState: Bundle)
    Init()
    Listners()
    onStart()
    onStop()
```

**ENTITY CLASSES ADMIN SIDE
(MODEL CLASSES)**

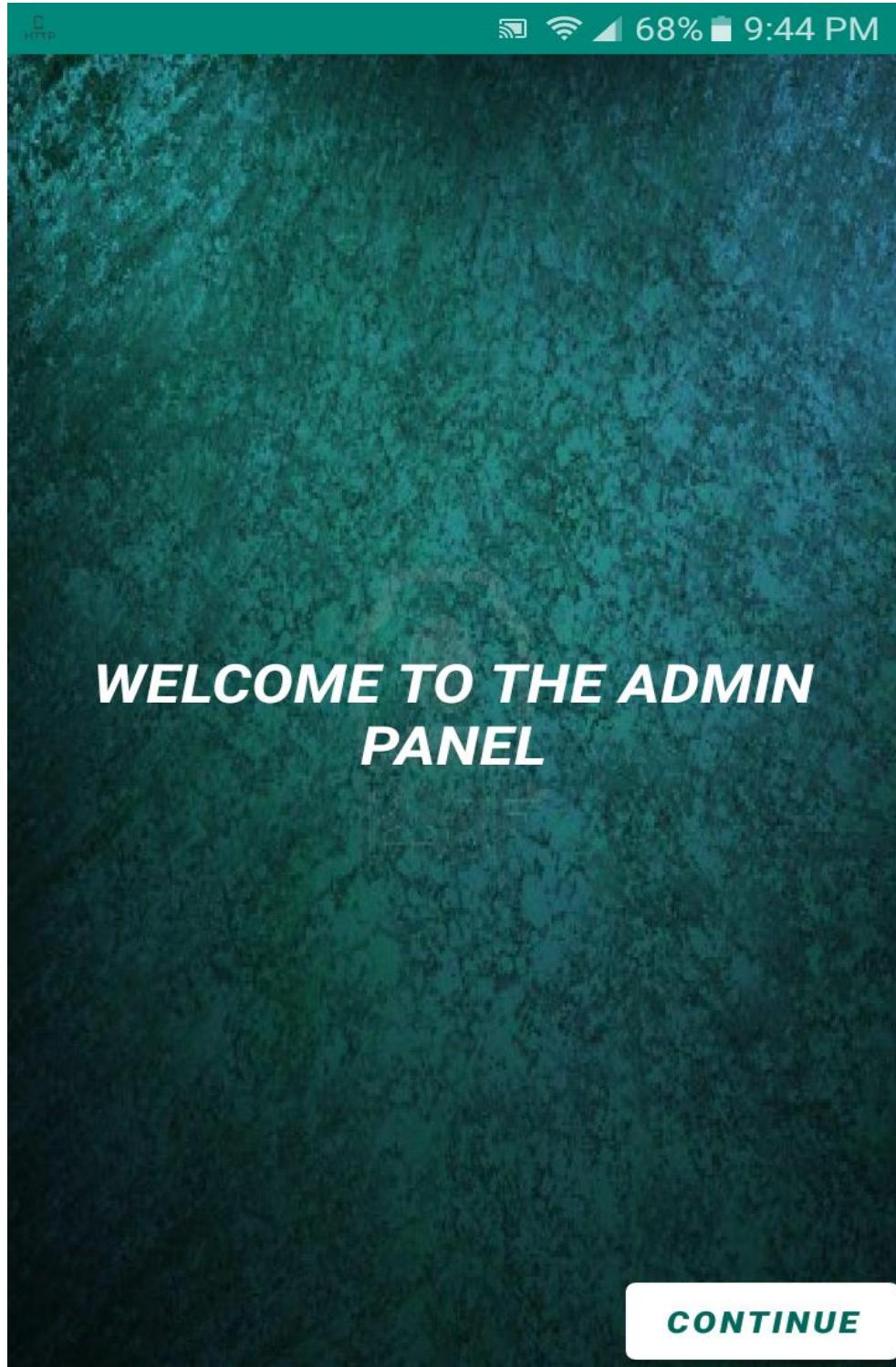
ADMINMODEL
fullName: String age: String email: String password: String
getAge() setAge() getfullName() setfullName() getEmail() setEmail() getPassword() setPassword()

THOUGHTMODEL
addedOn: String addedBy: String image: String
getImage() setImage()

CATEGORYMODEL
addedOn: String addedBy: String category: String
getCategory() setcategory()

BOOKPOSTMODEL
addedOn: String addedBy: String image: String author: String bookName: String BookPrice: String BookDescription: String BookCondition: String bookStatus: String bookEdition: String bookCategory: String getImage() setImage() getBookName() setBookName() getAuthor() setAuthor() getPrice() setPrice() getStatus() setStatus() getDescript() setDescript() getEdition() setEdition() getCondition() setCondition() getAddedOn() setAddedOn() getAddedBy() setAddedBy()

1.2 Results





68% 9:45 PM



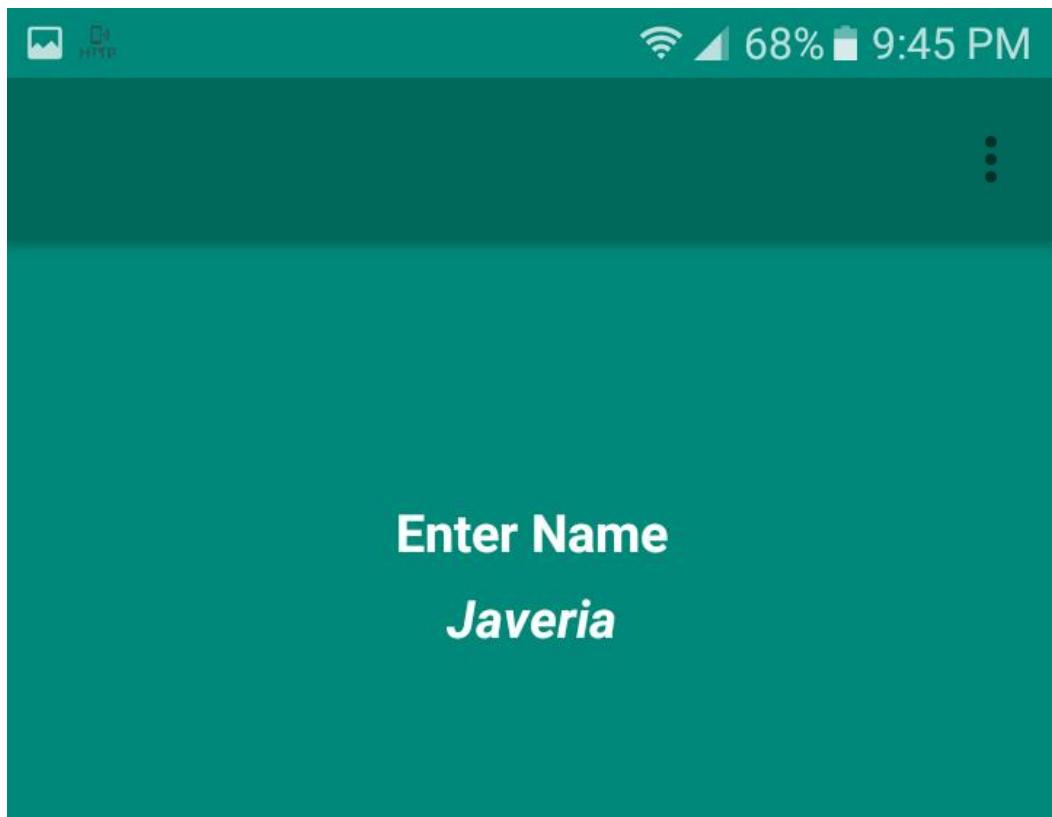
ADMIN PANEL

Login

Please wait..

[Forget Password?](#)

Login



hn34353435@gmail.com

Change Password

Save Changes

Profile Updated..



68% 9:46 PM



ADMIN PANEL

hn34353435@gmail.com

.....

Forgot Password?

Login



HTTP



68% 9:46 PM



Welcome Javeria !

Jan 02, 2021 21:46:15 PM



My Account



Add Book Category



**Add Thought of
the Day**



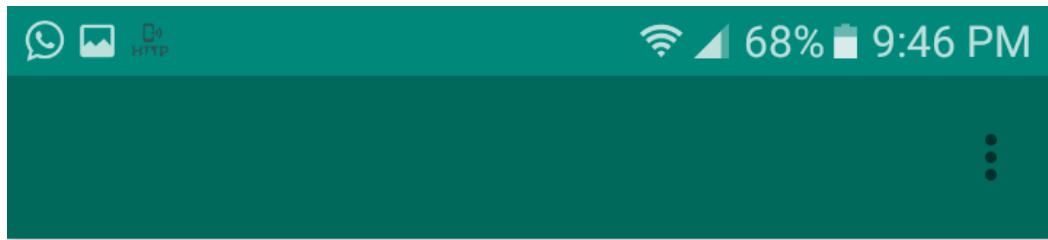
**Educational
Institutes Data**



**Books Posts
Request**



**Recommended
Data**



+ ADD NEW CATEGORY

English



Fiction



Medical



Others





68% 9:46 PM



Add Book Category

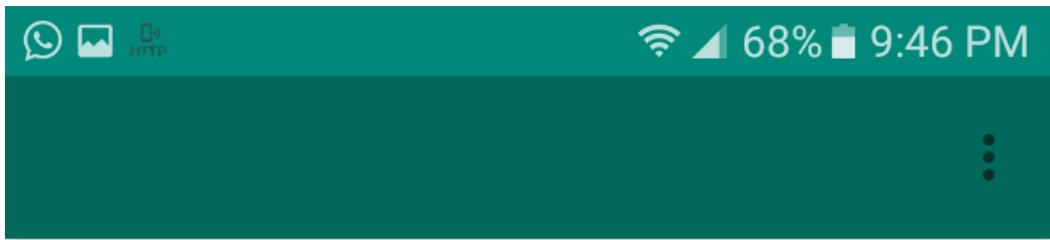
Enter New Book Category name

Mathematics|

Add New Category



Dear User, please wait while
we are adding the new
category.



+ ADD NEW CATEGORY

English



Fiction



Mathematics

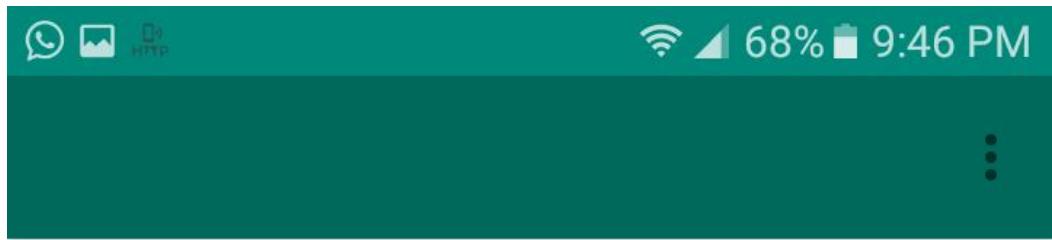


Medical



Others





+ ADD NEW IMAGE



Mistakes
are proof
that you
are trying.

A central image featuring a watercolor-style background with horizontal stripes in green, blue, and pink. Overlaid on this background is a white rectangular box containing a quote in a black, cursive font. The quote reads: "Mistakes are proof that you are trying."

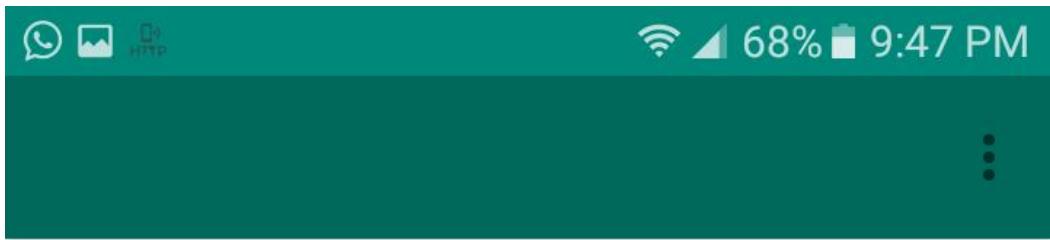


68% 9:47 PM

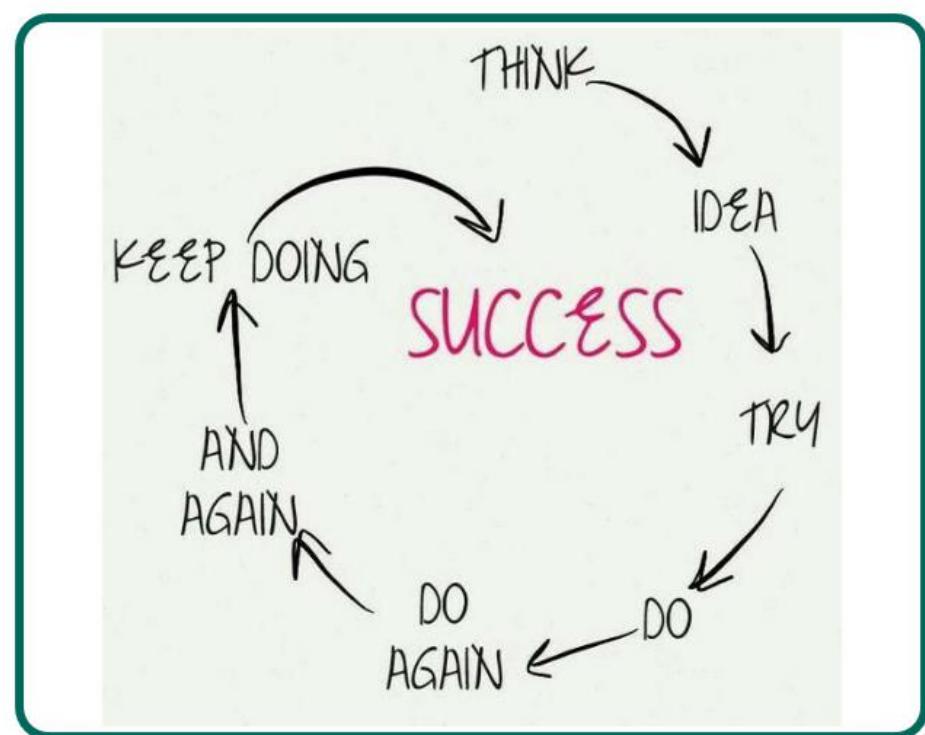


Add New Image

Dear User, please wait while
we are adding the new
Image.



+ ADD NEW IMAGE





HTTP



68%



9:47 PM

⋮

PUBLIC**PRIVATE**

Institute Name

Abdul Wali Khan University

Location

Mardan

Major Fields

Natural Sciences

Type

Public

Major Degree Program

BS Botany



HTTP



68%



9:47 PM

⋮

PUBLIC**PRIVATE**

Institute Name

The Aga Khan University

Location

Karachi

Major Fields

Medical & Health Sciences

Type

Private

Major Degree Program

MBBS



PENDING

ACCEPTED

REJECTED



Jannat K Pattay

VIEW

APPROVE

REJECT

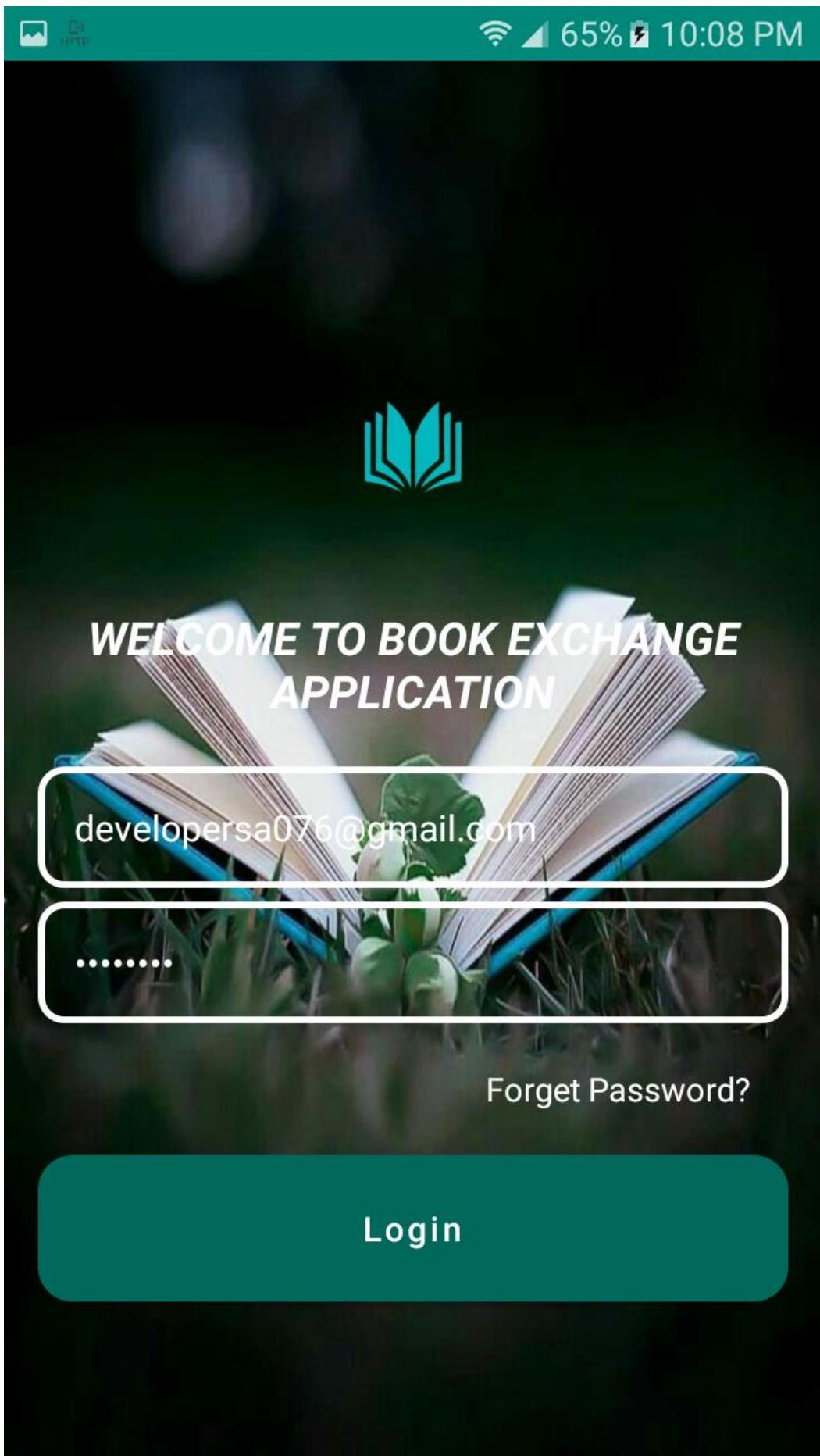


*Find, Discover, Choose and Buy
old books online.*



Join Now

Already have an Account? [Login](#)





HTTP



65%



10:09 PM



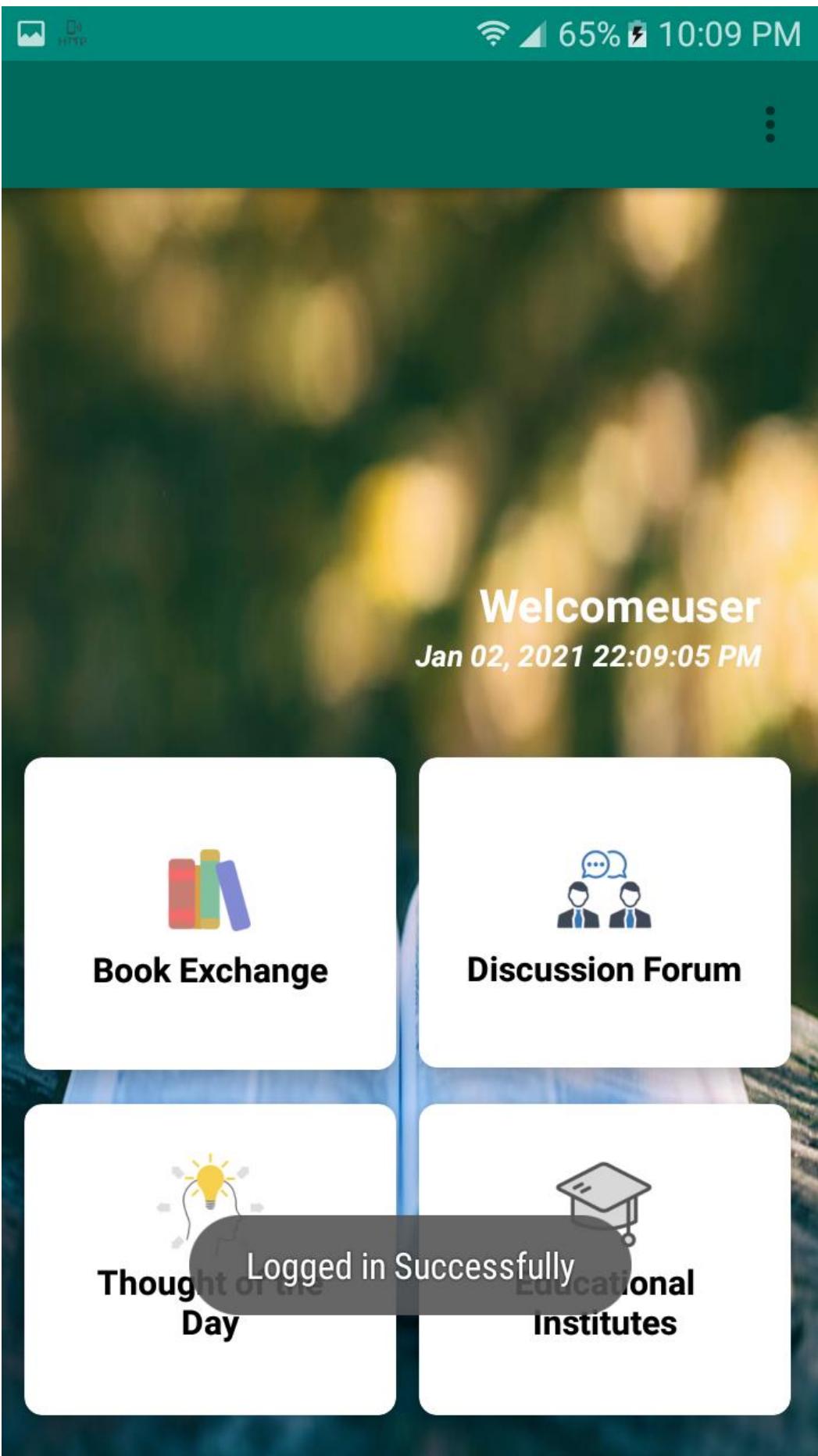
Book Exchange System



Email

Reset password





HTTP

65% 10:09 PM

⋮

 Book Posted By: Anabiya

Book Name
Jannat K Pattay

Category
Fiction

Class Level
Other

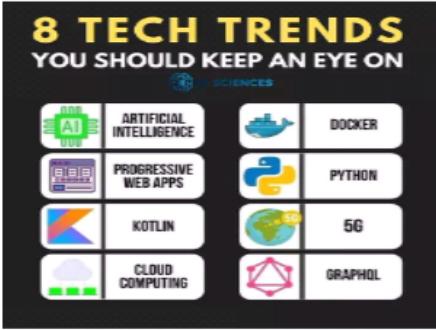
Price
700

[More Details](#) [BUY](#)

2 Share 1 Share

Like Comment Share

Like Magazine ...





Home



My chats



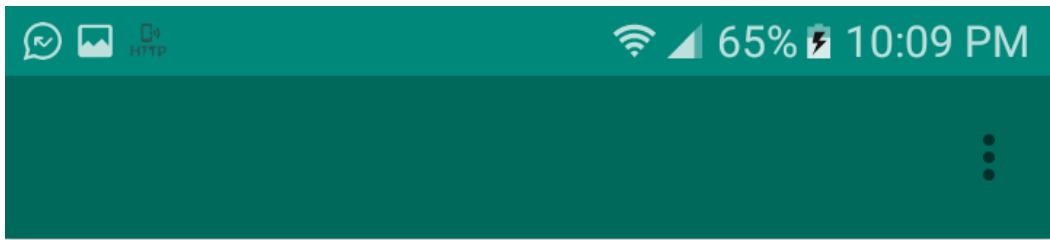
Sell



Notifications



My Ads



Add Book

Enter Book Name

Enter Book Name

Author Name

Enter Author Name

Choose Book Category



Select Class of your Book



Book Edition

Enter Edition

Book Condition



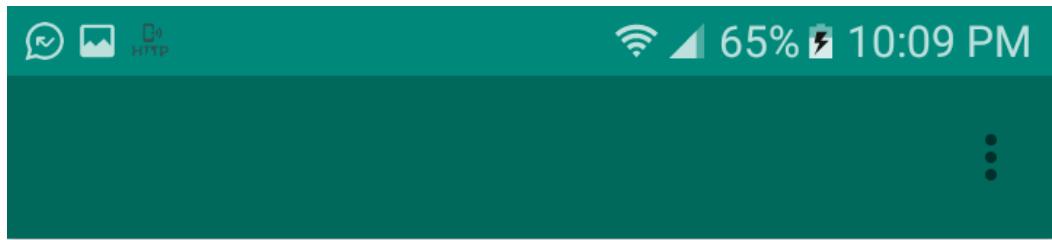
Book Price

Enter Book Price



Book Description

Enter Book Description



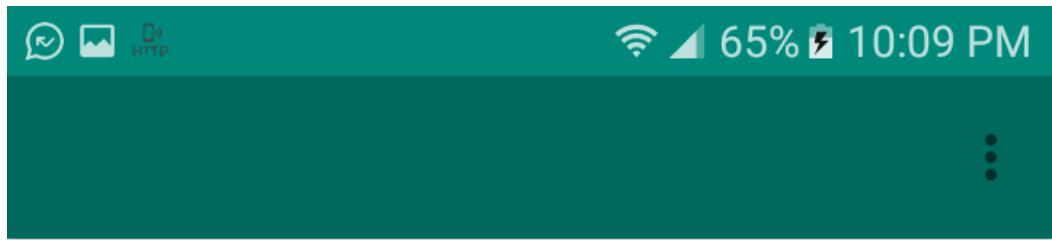
(11/06/2020)

Test User



hello

12:00AM



(11/06/2020)

Test User



hi

12:00AM

(11/06/2020)

Test User



hello

12:00AM





65% 10:09 PM



PUBLIC

PRIVATE

Institute Name

Abdul Wali Khan University

Location

Mardan

Major Fields

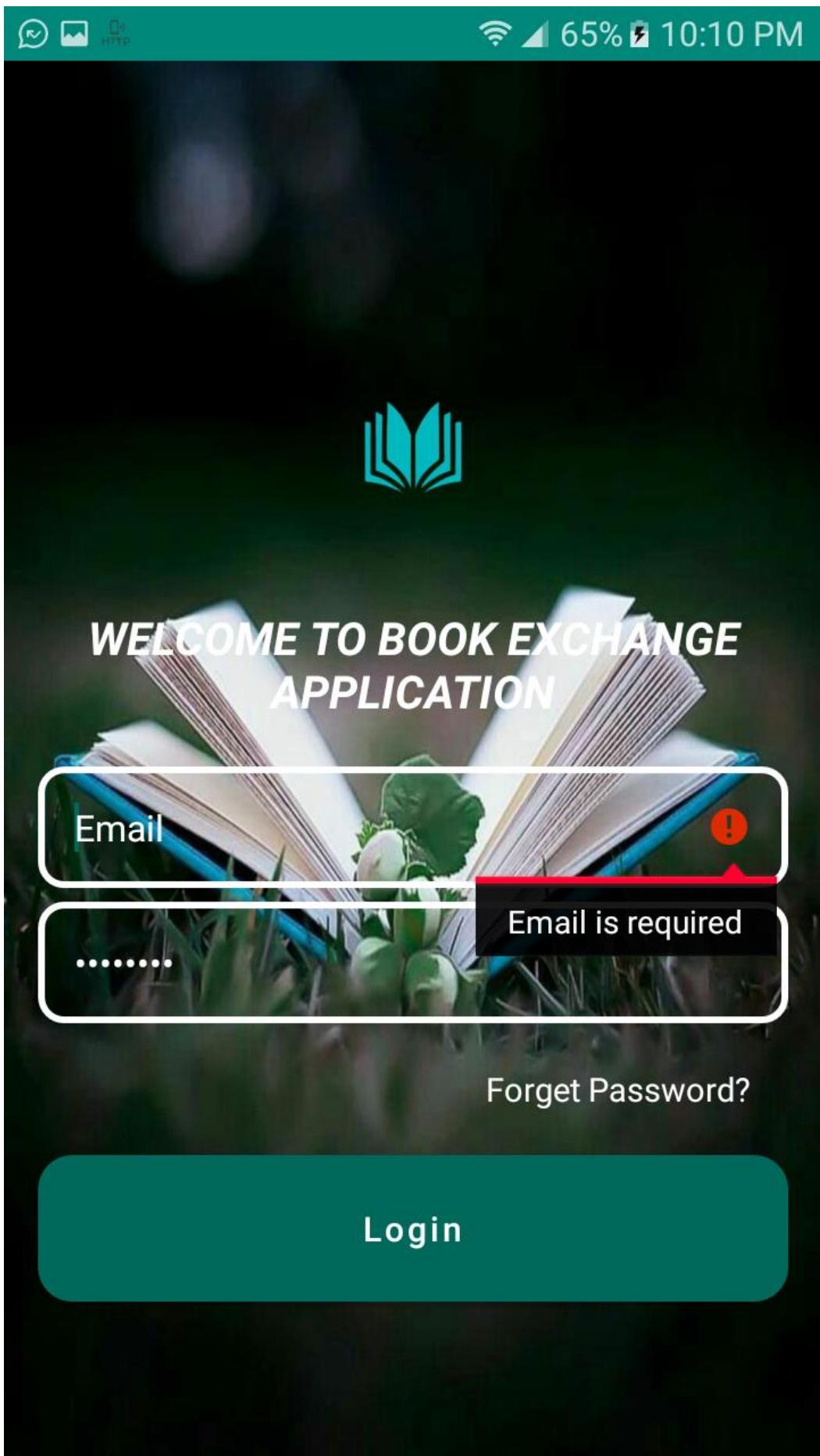
Natural Sciences

Type

Public

Major Degree Program

BS Botany

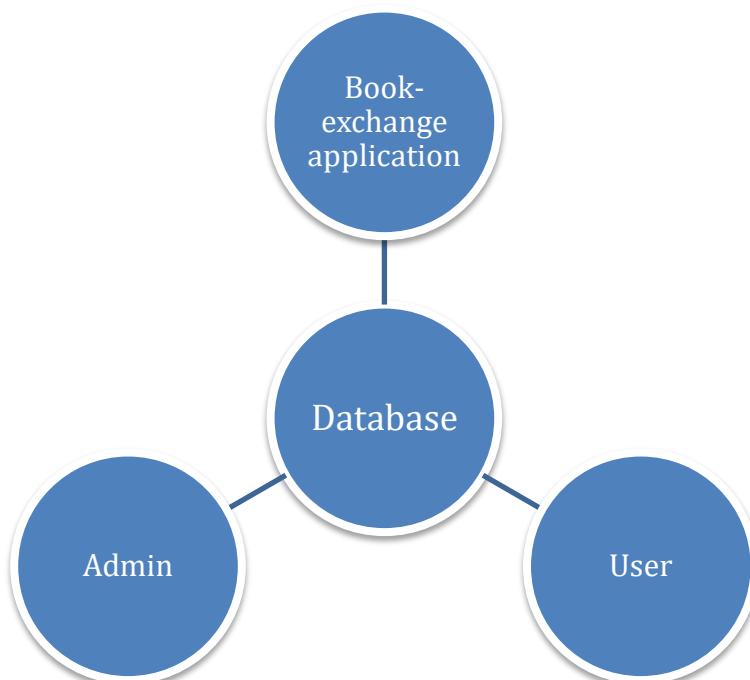


Overall Description

Product Perspective

This product was designed to assist those people who used to read books and they have a big collection of their old books using this application we enable them to sell their old books in reasonable price and get benefit through this application. However it could also be observed as an independent free-standing product and evaluate without depicting any differentiation.

The functionality of the product is to assist those students who want to get information about several universities of Karachi, provide platform to those who want to sell their books, motivate the user of the applications and help users by answering the queries related to studies and coding by providing useful information.



Product Functions

Book exchange Functions:

Book-exchange application Functions:

- Create Admin/Users Profiles
- Update profiles
- Watch my posted ads
- Chat of buyer and seller
- Exchanging old books
- Set query on discussion forum
- Viewing details of Universities that exist in Karachi
- Motivate users
- Resolving queries related to studies
- Logging of User ID with email and password authentication

User Classes and Characteristics

There are 3 basic user classes.

One is for Admin who has more rights than any other user and can insert/modify any kind of data. They have the best technical expertise and the highest privilege levels. They are the ones assigning roles and creating profiles and giving access to other users, adding thought images, new category of books, download csv reminders. They most frequently use the system but only a limited set of functions are their concern.

Students are the user class that needs to be satisfied in this regard. The Students can sell their old books and buy new books, see thought images and can chat on discussion forum.

The third user is data base which have to manage all data.

1.2.1 General Comments

As the characteristics narrate above, the precondition approved in the proposal have been fulfil to the prime of our attempt, the features of the project integrate to structure a powerful environment that is user-friendly and to come-up with a handy instinctive application. The graphical interface is accessible and relatively effort less to adapt. This application is easy to use because it is available at finger-tips.

3. Conclusion and Future Work

3.1. Conclusion

The whole time span of the project has been a significant learning chance with distinctive challenges that we strive to face head-on and with as much passion as we can manage. The acquirement of our stated objective has proved to be a crest of complete year's value of studying new things and making blunder, the end outcome is something which we are proud to present in front of the committee.

3.2. Recommendations/Suggestions of Future Work

The modular nature of the project allows for easy expansion, future work can include Real-time calling through this application for communicating to the sellers for buying an old book in the book exchange feature. The interactivity can also be enhanced by adding a feature in chat activity as recording audio so that buyer and seller can record their audios which save their time from typing the message. Additionally Sellers Recommendation system might also be added. In educational Institute feature we could work on crawler for fetching the images with the content. In thought of the day we can allow our user to post motivational images.

References

<https://www.youtube.com/watch?v=gPqJcPtN18I&list=PLxefhmF0pcPmtdoud8f64EpgapkclCllj>
https://www.youtube.com/watch?v=aS_9RbCyHg
<https://www.youtube.com/watch?v=oRkJvg5vCTk>
<https://www.youtube.com/watch?v=QOxqGzYGJXA>
<https://stackoverflow.com/questions/24438170/manifest-merger-failed-uses-sdkminsdkversion-14>
<https://stackoverflow.com/questions/52035063/android-resource-linking-failed>
[rd-in-firebase](https://stackoverflow.com/questions/15911165/create-an-empty-child-reco)
<https://medium.com/@hannah15198/convert-csv-to-json-with-python-b8899c722f6d>
<https://firebase.google.com/docs/firestore/manage-data/export-import>
[ragments](https://stackoverflow.com/questions/35533905/how-to-handle-onclick-in-f)
[utton-method](https://stackoverflow.com/questions/21192386/android-fragment-onclick-b)
[er-for-a-fragment-on-android](https://stackoverflow.com/questions/27728439/implementing-onclicklisten)
[tivity](https://stackoverflow.com/questions/20835933/intent-from-fragment-to-ac)
[gment](https://stackoverflow.com/questions/29243351/layoutinflater-inflate-fra)
<https://stackoverflow.com/questions/6495898/findviewbyid-in-fragment>
<https://www.youtube.com/watch?v=gtRQxYKLARG>
[https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))
https://en.wikipedia.org/wiki/Microsoft_Windows

[n-pandas/](https://www.geeksforgeeks.org/adding-new-column-to-existing-dataframe-i)
<https://rubikscode.net/2019/12/02/scraping-images-with-python/>
<https://www.iditect.com/how-to/55372694.html>
[-to-repeat-at-position-0/46103801](https://stackoverflow.com/questions/46103756/python-regex-error-nothing)
[eb-using-beautifulsoup-206a7633e948](https://towardsdatascience.com/a-tutorial-on-scraping-images-from-the-w)
<https://merafuture.pk/PakistaniUniversitiesPakistaniDegrees>
[-firebase-storage](https://stackoverflow.com/questions/39702304/retrieve-stored-image-from)
[-in-array-in-android](https://stackoverflow.com/questions/47370834/getting-data-from-firebase)
[base-database-and-storing-in-an-array-list](https://stackoverflow.com/questions/41587829/retrieving-data-from-fir)
[mages+from+firebase+in+android+studio](https://www.youtube.com/results?search_query=how+to+retrieve+multiple+i)
<https://www.iconfinder.com/>
<https://www.youtube.com/watch?v=OVSe6bAnjgg>
<https://www.geeksforgeeks.org/bottom-navigation-bar-in-android/>
[m-navigation-view-not-showing-item-in-my-menu-added](https://stackoverflow.com/questions/58513786/solvedandroid-studio-botto)
[ivity/](https://www.androidlearning.com/moving-from-one-activity-to-another-act)
https://en.wikipedia.org/wiki/Android_Studio

https://www.youtube.com/watch?v=T7CQzCRPiQU&list=PLxefhmF0pcPlqmH_VfWneUjfuhreUz-O&index=46
<https://www.youtube.com/watch?v=ziZvfckxig0>
https://www.youtube.com/watch?v=hLH2F3Qro_Y&list=PLxefhmF0pcPlqmH_VfWneUjfuhreUz-O&index=2
<https://www.youtube.com/playlist?list=PLxefhmF0pcPn6o31QZHto6CW6Kfnu4Eqv>
https://www.youtube.com/watch?v=6keVIot98QU&list=PLxefhmF0pcPlqmH_VfWneUjfuhreUz-O

Appendix A – Development

Environment

- **Operating Systems(s)**
 - **Windows**
 - **Microsoft Windows**, commonly referred to as **Windows**, is a group of several proprietary graphical operating system families, all of which are developed and marketed by Microsoft. Each family caters to a certain sector of the computing industry. Active Microsoft Windows families include Windows NT and Windows IoT; these may encompass subfamilies, (e.g. Windows Server or Windows Embedded Compact) (Windows CE). Defunct Microsoft Windows families include Windows 9x, Windows Mobile and Windows Phone.
 - Microsoft introduced an operating environment named *Windows* on November 20, 1985, as a graphical operating system shell for MS-DOS in response to the growing interest in graphical user interfaces (GUIs).
 - *Developer* – Microsoft
Version – Windows 10
Environment – 64 Bit
Website – windows.microsoft.com

Programming Language(s)



Java is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let application developers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can

run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle) and released in 1995 as a core component of Sun Microsystems' Java platform.

Java programming language logo.svg
Paradigm Multi-paradigm: generic, object-oriented (class-based), imperative, reflective
Designed by James Gosling
Developer Oracle Corporation et al.
First appeared May 23, 1995; 25 years ago[1]
Stable release
Java SE 15[2] / September 15, 2020; 3 months ago
Typing discipline Static, strong, safe, nominative, manifest
Filename extensions .java, .class, .jar
Website oracle.com/java/

Object-oriented programming

Object-oriented programming (OOP) is a programming paradigm that represents the concept of "objects" that have data fields (attributes that describe the object) and associated procedures known as methods. Objects, which are usually instances of classes, are used to interact with one another to design applications and computer programs.

Object-oriented analysis and design

Object Oriented Analysis & Design (OOAD) is a popular technical approach to analyzing, **designing** an application, system, or business by applying The **object-oriented** paradigm and visual modeling throughout the development life cycles to foster better stakeholder communication and product quality.

Communication and Leadership

Democratic Decentralized

The crucial delegation of power among group mates remains that power is equally distributed as well as the responsibility. We both had the veto power to decide upon the development processes and methodologies and the functions and modules were distributed while we worked separately on them with the consensus and co-operation of our partner. The power did not rest with any one particular individual and overall it was a constructive and to-and-fro development.

Appendix B – Tools and Software

Android Studio



Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on Jet Brains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio 4.1 running on Linux

Developer(s) Google, Jet Brains

Stable release 4.1.1 (November 10, 2020; 52 days ago)

Preview release 4.2 Beta 2 (December 14, 2020; 18 days ago)

Written in Java, Kotlin and C++

Operating system Windows, macOS, Linux, Chrome OS

Size 727 to 877 MB

Type Integrated development environment (IDE)

License Freeware +Source code

Website developer.android.com/studio/index.html



Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011.^[1] In 2014, Google acquired the platform^[2] and it is now their flagship offering for app development.

Computing and development tools

Product(s) Building

Cloud Firestore, Firebase, MLCloud, Functions, Authentication, Hosting, Cloud Storage, Realtime Database

Quality

Crashlytics, Performance, Monitoring, Test, Lab, App Distribution

Business

In-App Messaging, Google, Analytics, Predictions, A/B Testing, Cloud Messaging, Remote Configuration, Dynamic Links

Leading companies: Google

Year of invention: 2012–present

Industrial sector(s): Computing and development tools

Founded: 2010

Founders: James Tamplin, Andrew Lee

Subsidiaries: Crashlytics, Fabric

Appendix C – Code

ADMIN SIDE CODE MODEL CLASSES

ADMINMODEL CLASS.java

```
package com.squares.adminapp.Models;

public class AdminModel {

    public String email,password,fullName,age;
    public AdminModel(){}

    public AdminModel(String email, String password) {
        this.email = email;
        this.password = password;
    }

    public String getAge() {
        return age;
    }

    public void setAge(String age) {
        this.age = age;
    }

    public AdminModel(String age, String email, String fullName) {
        this.age=age;
        this.email = email;
        this.password = fullName;
    }

    public String getFullName() {
        return fullName;
    }

    public void setFullName(String fullName) {
        this.fullName = fullName;
    }

    public String getEmail() {
        return email;
    }

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

    public String getPassword() {
        return password;
    }
}
```

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

BOOKPOST MODEL.java

```
package com.squares.adminapp.Models;

import android.net.Uri;

public class BooksPostModel {
    String added_by;
    long added_on;
    String author;
    String bookGrade;
    String bookName;
    String category;
    String condition;
    String description;
    String edition;
    String image;
    String pid;
    String price;
    String status;
    public BooksPostModel(){}
    public BooksPostModel(
        String added_by,
        long added_on,
        String author,
        String bookGrade,
        String bookName,
        String category,
        String condition,
        String description,
        String edition,
        String image,
        String pid,
        String price,
        String status
    ) {
        this.added_by = added_by;
        this.added_on = added_on;
        this.author = author;
        this.bookGrade = bookGrade;
        this.bookName = bookName;
        this.category = category;
        this.condition = condition;
        this.description = description;
        this.edition = edition;
        this.image = image;
        this.pid = pid;
        this.price = price;
        this.status = status;
    }
    public String getImage(){
        return this.image;
    }
}
```

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

public Uri getImageUri(){
    return Uri.parse(this.image);
}

public String getBookName(){
    return this.bookName;
}

public void setBookName(String bookName){
    this.bookName = bookName;
}
public String getAdded_by(){
    return this.added_by;
}

public void setAdded_by(String added_by){
    this.added_by = added_by;
}

public long getAdded_on(){
    return this.added_on;
}

public void setAdded_on(long added_on){
    this.added_on = added_on;
}
public String getStatus(){
    return this.status;
}

public void setStatus(String status){
    this.status = status;
}

public String getAuthor(){
    return this.author;
}

public void setAuthor(String author){
    this.author = author;
}

public String getBookGrade(){
    return this.bookGrade;
}

public void setBookGrade(String bookGrade){
    this.bookGrade = bookGrade;
}

public String getCategory(){
    return this.category;
}
```

```

    }

    public void setCategory(String category){
        this.category = category;
    }

    public String getCondition(){
        return this.condition;
    }

    public void setCondition(String condition){
        this.condition = condition;
    }

    public String getDescription(){
        return this.description;
    }

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

    public String getEdition(){
        return this.edition;
    }

    public void setEdition(String edition){
        this.edition = edition;
    }

    public String getPid(){
        return this.pid;
    }

    public void setPid(String pid){
        this.pid = pid;
    }

    public String getPrice(){
        return this.price;
    }

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

```

CATEGORYMODEL.java

```

package com.squares.adminapp.Models;

public class CategoryModel {
    String category,added_by;
    long added_on;
    public CategoryModel()
}

```

```

    {
}

public CategoryModel(String added_by, long added_on, String category) {
    this.category = category;
    this.added_on = added_on;
    this.added_by = added_by;
}

public String getCategory() {
    return this.category;
}

public void setCategory(String category) {
    this.category = category;
}

public long getAdded_on() {
    return added_on;
}

public void setAdded_on(long added_on) {
    this.added_on = added_on;
}

public String getAdded_by() {
    return added_by;
}

public void setAdded_by(String added_by) {
    this.added_by = added_by;
}
}

```

EDUCATIONALINSTITUTEMODEL.java

```

package com.squares.adminapp.Models;

import android.net.Uri;

public class EducationalInstituteModel {

    String Image, InstituteName, Location, Type, MajorProgram, MajorFieis;

    public EducationalInstituteModel() {
    }

    public EducationalInstituteModel(String image, String instituteName, String location, String type, String majorProgram, String majorFieis) {
        Image = image;
        InstituteName = instituteName;
        Location = location;
        Type = type;
        MajorProgram = majorProgram;
        MajorFieis = majorFieis;
    }

    public String getImage() {
        return Image;
    }
}

```

```

public void setImage(String image) {
    Image = image;
}

public String getInstituteName() {
    return InstituteName;
}

public void setInstituteName(String instituteName) {
    InstituteName = instituteName;
}

public String getLocation() {
    return Location;
}

public void setLocation(String location) {
    Location = location;
}

public String getType() {
    return Type;
}

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

public String getMajorProgram() {
    return MajorProgram;
}

public void setMajorProgram(String majorProgram) {
    MajorProgram = majorProgram;
}

public String getMajorFieis() {
    return MajorFieis;
}

public void setMajorFieis(String majorFieis) {
    MajorFieis = majorFieis;
}
}

```

THOUGHTSMODEL.java

```

package com.squares.adminapp.Models;

import android.net.Uri;

public class ThoughtModel {
    String image;

    public ThoughtModel() {
    }

    public ThoughtModel(String image) {
        this.image = image;
    }
}

```

```

        public String getImage() {
            return image;
        }

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

        public String getThoughtImageUrl() {
            return image;
        }

    }

```

USERSMODEL.java

```

package com.squares.adminapp.Models;

import android.net.Uri;

public class UsersModel {

    public String fullName,
    email,contact,country,city,UserProfileImage,password,added_on,updated_on;
    public boolean isSeller, isBuyer;
    public UsersModel()
    {}
    public Uri getImageUri(){
        return Uri.parse(this.UserProfileImage);
    }

    public UsersModel(String fullName, String email, String contact, String password)
    {
        this.fullName = fullName;
        this.email = email;
        this.contact = contact;
        this.password=password;
    }

    public UsersModel(String fullName, String email, String contact, String country,
    String city, String userProfileImage, String updated_on) {
        this.fullName = fullName;
        this.email = email;
        this.contact = contact;
        this.country = country;
        this.city = city;
        this.UserProfileImage = userProfileImage;
        this.updated_on=updated_on;
    }

    public String getFullName() {
        return fullName;
    }

    public void setFullName(String fullName) {
        this.fullName = fullName;
    }
}

```

```
public String getEmail() {
    return email;
}

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

public String getContact() {
    return contact;
}

public void setContact(String contact) {
    this.contact = contact;
}

public String getCountry() {
    return country;
}

public void setCountry(String country) {
    this.country = country;
}

public String getCity() {
    return city;
}

public void setCity(String city) {
    this.city = city;
}

public String getUserProfileImage() {
    return UserProfileImage;
}

public void setUserProfileImage(String userProfileImage) {
    UserProfileImage = userProfileImage;
}

public String getPassword() {
    return password;
}

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

public String getAdded_on() {
    return added_on;
}

public void setAdded_on(String added_on) {
    this.added_on = added_on;
}

public String getUpdated_on() {
    return updated_on;
}
```

```

        public void setUpdated_on(String updated_on) {
            this.updated_on = updated_on;
        }

        public boolean isSeller() {
            return isSeller;
        }

        public void setSeller(boolean seller) {
            isSeller = seller;
        }

        public boolean isBuyer() {
            return isBuyer;
        }

        public void setBuyer(boolean buyer) {
            isBuyer = buyer;
        }

        public UsersModel(String fullName, String email, String contact, String password,
String country, String city, String userProfileImage, String added_on, String
updated_on, boolean isSeller, boolean isBuyer) {
            this.fullName = fullName;
            this.email = email;
            this.contact = contact;
            this.country = country;
            this.city = city;
            UserProfileImage = userProfileImage;
            this.password = password;
            this.added_on=added_on;
            this.updated_on=updated_on;
            this.isSeller = isSeller;
            this.isBuyer = isBuyer;
        }

        public UsersModel(String fullName, String email, String contact, String password,
String country, String city, String userProfileImage, String updated_on) {
            this.fullName = fullName;
            this.email = email;
            this.contact = contact;
            this.country = country;
            this.city = city;
            UserProfileImage = userProfileImage;
            this.password = password;
            this.added_on=updated_on;
        }
    }
}

```

ADMIN ADAPTER CLASSES

BookPostAdapter.java

```

package com.squares.adminapp.Adapters;

import android.app.AlertDialog;
import android.content.Intent;
import android.net.Uri;
import android.util.Log;

```

```

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

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

import com.bumptech.glide.Glide;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.adminapp.AddBookCategoryActivity;
import com.squares.adminapp.Models.BooksPostModel;
import com.squares.adminapp.Models.CategoryModel;
import com.squares.adminapp.R;
import com.squares.adminapp.ShowBookCategoriesActivity;

import de.hdodenhof.circleimageview.CircleImageView;

public class BooksPostAdapter extends FirebaseRecyclerAdapter<BooksPostModel,
BooksPostAdapter.myviewholder> {
    public BooksPostAdapter(@NonNull FirebaseRecyclerOptions<BooksPostModel>
options) {
        super(options);

    }
    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull BooksPostModel model)
    {
//        Uri img = model.getImageUri();
        String title = model.getBookName();

Glide.with(holder.bookImage.getContext()).load(model.getImageUri()).into(holder.bo
okImage);
        holder.title.setText(title);

        holder.approveBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                DatabaseReference currentBookRef =
FirebaseDatabase.getInstance().getReference("Books").child(model.getPid());
                currentBookRef.child("status").setValue("APPROVED", new
DatabaseReference.CompletionListener() {
                    @Override
                    public void onComplete(@Nullable DatabaseError error, @NonNull
DatabaseReference ref) {
                        if(error != null) {
                            Log.v("Error", "There is an error " +
error.getMessage());
                        } else {
                            Log.v("Success", model.getBookName() + " has been
approved");
                        }
                    }
                });
            }
        });
    }
}

```

```

        }
    });
});

holder.rejectBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        DatabaseReference currentBookRef =
        FirebaseDatabase.getInstance().getReference("Books").child(model.getPid());
        currentBookRef.child("status").setValue("DECLINED", new
        DatabaseReference.CompletionListener() {
            @Override
            public void onComplete(@Nullable DatabaseError error, @NonNull
        DatabaseReference ref) {
                if(error != null) {
                    Log.v("Error", "There is an error " +
error.getMessage());
                } else {
                    Log.v("Success", model.getBookName() + " has been
approved");
                }
            }
        });
    }
});

@NonNull
@Override
public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
{
    View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_admin_book_post
ed_for_approvel,parent,false);
    return new BooksPostAdapter.myviewholder(view);
}

static class myviewholder extends RecyclerView.ViewHolder
{
    CircleImageView bookImage;
    TextView title;
    Button approveBtn;
    Button rejectBtn;

    public myviewholder(@NonNull View itemView)
    {
        super(itemView);
        bookImage = itemView.findViewById(R.id.user_request_book_post_image);
        title = itemView.findViewById(R.id.user_request_book_posted_by);
        approveBtn = itemView.findViewById(R.id.post_approve_button);
        rejectBtn = itemView.findViewById(R.id.post_cancel_button);

    }
}

```

```
}
```

CategoryAdapter.java

```
package com.squares.adminapp.Adapters;

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

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

import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.squares.adminapp.Models.CategoryModel;
import com.squares.adminapp.R;

public class CategoryAdapter extends
FirebaseRecyclerAdapter<CategoryModel,CategoryAdapter.myviewholder> {

    public CategoryAdapter(@NonNull FirebaseRecyclerOptions<CategoryModel> options) {
        super(options);
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull CategoryModel model)
    {
        holder.CatnameTv.setText(model.getCategory());
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_show_book_categories,parent,false);
        return new myviewholder(view);
    }

    static class myviewholder extends RecyclerView.ViewHolder
    {
        CardView ThoughtImagesCard;
        ImageView RemoveCatImagesIv;
        TextView CatnameTv;
        public myviewholder(@NonNull View itemView)
        {
            super(itemView);
            ThoughtImagesCard=(CardView) itemView.findViewById(R.id.thoughtCardView);
            RemoveCatImagesIv=(ImageView) itemView.findViewById(R.id.removeCat);
        }
    }
}
```

```
        CatnameTv=(TextView) itemView.findViewById(R.id.catNameTv);
    }
}
```

EducationalInstituteAdapter.java

```
package com.squares.adminapp.Adapters;

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

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

import com.bumptech.glide.Glide;
import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.adminapp.Models.BooksPostModel;
import com.squares.adminapp.Models.EducationalInstituteModel;
import com.squares.adminapp.R;

import de.hdodenhof.circleimageview.CircleImageView;

public class EducationalInstituteAdapter extends
FirebaseRecyclerAdapter<EducationalInstituteModel,
EducationalInstituteAdapter.myviewholder> {

    public EducationalInstituteAdapter(@NonNull
FirebaseRecyclerOptions<EducationalInstituteModel> options) {
        super(options);
    }

    @Override
    protected void onBindViewHolder(@NonNull
EducationalInstituteAdapter.myviewholder holder, int position, @NonNull
EducationalInstituteModel model)
    {
    //
        holder.InstituteNameTv.setText(model.getInstituteName());
        holder.LocationTv.setText(model.getLocation());
        holder.MajorFieldsTv.setText(model.getMajorFiebs());
        holder.TypeTv.setText(model.getType());
        holder.MajorProgramTv.setText(model.getMajorProgram());
    }

    @NonNull
    @Override
```

```

    public EducationalInstituteAdapter.myviewholder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType)
{
    View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_view_educationa
l_institute,parent,false);
    return new EducationalInstituteAdapter.myviewholder(view);
}

static class myviewholder extends RecyclerView.ViewHolder
{

    TextView InstituteNameTv,LocationTv, MajorFieldsTv,TypeTv,MajorProgramTv;

    public myviewholder(@NonNull View itemView)
    {
        super(itemView);
        InstituteNameTv = itemView.findViewById(R.id.InstituteNameRetrieve);
        LocationTv = itemView.findViewById(R.id.InstituteLocationTv);
        MajorFieldsTv = itemView.findViewById(R.id.majorFields);
        TypeTv = itemView.findViewById(R.id.TypeTv);
        MajorProgramTv = itemView.findViewById(R.id.MajorProgramTv);

    }
}
}

```

ThoughtImagesAdapter.java

```

package com.squares.adminapp.Adapters;

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

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

import com.bumptech.glide.Glide;
import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.squares.adminapp.Models.ThoughtModel;
import com.squares.adminapp.R;
import com.squareup.picasso.Picasso;

import java.util.List;

public class ThoughtImagesAdapter extends FirebaseRecyclerAdapter<ThoughtModel,
ThoughtImagesAdapter.myviewholder>
{

    public ThoughtImagesAdapter(@NonNull FirebaseRecyclerOptions<ThoughtModel>
options) {

```

```

        super(options);
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull ThoughtModel model) {

Glide.with(holder.img.getContext()).load(model.getThoughtImageUrl()).into(holder.i
mg);
}

@NonNull
@Override
public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
{
    View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_show_thought_of
_the_day,parent,false);
    return new myviewholder(view);
}

class myviewholder extends RecyclerView.ViewHolder
{
    ImageView img;
    public myviewholder(@NonNull View itemView)
    {
        super(itemView);
        img=(ImageView) itemView.findViewById(R.id.thoughtImagesRetrieve);

    }
}
}

```

UsersAdapter.java

```

package com.squares.adminapp.Adapters;

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

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

import com.bumptech.glide.Glide;
import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.squares.adminapp.Models.UsersModel;
import com.squares.adminapp.R;

import de.hdodenhof.circleimageview.CircleImageView;

public class UsersAdapter extends

```

```

FirebaseRecyclerAdapter<UsersModel, UsersAdapter.myviewholder>
{
    public UsersAdapter(@NonNull FirebaseRecyclerOptions<UsersModel> options) {
        super(options);
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull UsersModel model)
    {
        holder.name.setText(model.getFullName());
        holder.added_on.setText(model.getAdded_on());
        holder.email.setText(model.getEmail());

        Glide.with(holder.img.getContext()).load(model.getImageUri()).into(holder.img);
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_users_cardview,
parent,false);
        return new myviewholder(view);
    }

    class myviewholder extends RecyclerView.ViewHolder
    {
        CircleImageView img;
        TextView name,added_on,email;
        public myviewholder(@NonNull View itemView)
        {
            super(itemView);
            img=(CircleImageView)itemView.findViewById(R.id.profileImage);
            name=(TextView)itemView.findViewById(R.id.UserNameTxt);
            added_on=(TextView)itemView.findViewById(R.id.UserAddedOnTv);
            email=(TextView)itemView.findViewById(R.id.UserEmailTxt);
        }
    }
}

```

ADMIN ACTIVITY CLASSES

AcceptedBookPostFragmentActivity

```

package com.squares.adminapp;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.FirebaseDatabase;

```

```

import com.squares.adminapp.Adapters.BooksPostAdapter;
import com.squares.adminapp.Models.BooksPostModel;

/**
 * A simple {@link Fragment} subclass.
 * Use the {@link AcceptedBookPostsFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class AcceptedBookPostsFragment extends Fragment {
    RecyclerView postsRecyclcerView;
    Button ViewPost;
    BooksPostAdapter adapter;

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

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

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

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

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View root=inflater.inflate(R.layout.fragment_accepted_book_posts,

```

```

        container, false);

        postsRecyclerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);
        ViewPost=(Button)root.findViewById(R.id.post_view_button);

        postsRecyclerView.setLayoutManager(new
LinearLayoutManager(getContext()));

        FirebaseRecyclerOptions<BooksPostModel> options =
            new FirebaseRecyclerOptions.Builder<BooksPostModel>()
.setQuery(FirebaseDatabase.getInstance().getReference("Books").orderByChild("status")
.equalTo("APPROVED"), BooksPostModel.class)
.build();

        adapter=new BooksPostAdapter(options);
        postsRecyclerView.setAdapter(adapter);

//        ViewPost.setOnClickListener(new View.OnClickListener() {
//            @Override
//            public void onClick(View view) {
//                Intent intent=new Intent(getActivity(),ViewFullBookPost.class);
//                startActivity(intent);
//            }
//        });

        return root;
    }

    @Override
    public void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    public void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

AddBookCategoryActivity

```

package com.squares.adminapp;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

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

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;

```

```

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.HashMap;

public class AddBookCategoryActivity extends AppCompatActivity {

    EditText AddNewCategoryEt;
    Button AddNewCatBtn;
    String BookCategoryName;
    private ProgressDialog loadingBar;
    Calendar calendar;
    private String currentDate, currentTime, added_by, productRandomKey;
    private long added_on;
    FirebaseUser current_user;
    private DatabaseReference BookCatRef;

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

        init();
        listeners();
    }

    private void listeners() {
        AddNewCatBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AddBookCategory();
            }
        });
    }

    private void AddBookCategory() {
        BookCategoryName = AddNewCategoryEt.getText().toString().trim();
        added_on = new Date().getTime();

        if(BookCategoryName.isEmpty())
        {
            AddNewCategoryEt.setError("Book category is required");
            AddNewCategoryEt.requestFocus();
        }
        else
        {
            loadingBar.setTitle("Add New Category");
            loadingBar.setMessage("Dear User, please wait while we are adding the new category.");
            loadingBar.setCanceledOnTouchOutside(false);
            loadingBar.show();
        }
    }
}

```

```

HashMap<String, Object> bookMap = new HashMap<>();

bookMap.put("category", BookCategoryName);
bookMap.put("added_on", added_on);
bookMap.put("added_by", added_by);
productRandomKey = BookCategoryName.toLowerCase();
BookCatRef.child(productRandomKey)
.addListenerForSingleValueEvent(
    new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            if(dataSnapshot.getValue() != null) {
                Toast.makeText(AddBookCategoryActivity.this,
"Error: This Category already exists!", Toast.LENGTH_SHORT).show();
                loadingBar.dismiss();
                return;
            }
        }

        BookCatRef.child(productRandomKey).updateChildren(bookMap)
            .addOnCompleteListener(new
OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void>
task)
                {
                    if (task.isSuccessful())
                    {
                        finish();
                        loadingBar.dismiss();

                        Toast.makeText(AddBookCategoryActivity.this, "Category Added",
Toast.LENGTH_SHORT).show();
                    }
                    else
                    {
                        loadingBar.dismiss();
                        String message =
task.getException().toString();

                        Toast.makeText(AddBookCategoryActivity.this, "Error: " + message,
Toast.LENGTH_SHORT).show();
                    }
                }
            });
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {
            loadingBar.dismiss();
            String message = databaseError.getMessage();
            Toast.makeText(AddBookCategoryActivity.this, "Error: " +
message, Toast.LENGTH_SHORT).show();
        }
    });

```

```

        }

    }

    private void init() {
        AddNewCategoryEt=(EditText) findViewById(R.id.addNewCatEt);
        AddNewCatBtn=(Button) findViewById(R.id.AddNewCategoryBtn);
        loadingBar=new ProgressDialog(this);

        current_user=FirebaseAuth.getInstance().getCurrentUser();
        assert current_user != null;
        added_by=current_user.getUid();

        BookCatRef = FirebaseDatabase.getInstance().getReference("bookCategories");

        calendar = Calendar.getInstance();

        SimpleDateFormat _currentDate = new SimpleDateFormat("MMM dd, yyyy");
        currentDate = _currentDate.format(calendar.getTime());

        SimpleDateFormat _currentTime = new SimpleDateFormat("HH:mm:ss a");
        currentTime = _currentTime.format(calendar.getTime());
    }
}

```

AddThoughtImagesActivity

```

package com.squares.adminapp;

import android.app.AlertDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

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

import com.google.android.gms.tasks.Continuation;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.storage.FirebaseStorage;
import com.google.firebaseio.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

```

```

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.HashMap;

public class AddThoughtImages extends AppCompatActivity {
    private Uri ImageUri;
    private static final int GalleryPick=1;
    private ProgressDialog loadingBar;
    private String ImageRandomKey, downloadImageUrl;
    private StorageReference ThoughtImagesRef;
    private DatabaseReference ThoughtRef;
    Calendar calendar;
    ImageView thoughtImageView;
    Button UploadBtn;
    String added_on;
    String added_by;
    String saveCurrentDate;
    String saveCurrentTime;
    FirebaseUser current_user;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_admin_upload_thought_images);
        init();
        listeners();
    }

    private void listeners() {
        thoughtImageView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                OpenGallery();
            }
        });
        UploadBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                ValidateImageEnterData();
            }
        });
    }
    private void ValidateImageEnterData()
    {

        if( ImageUri == null) {
            thoughtImageView.requestFocus();
            Toast.makeText(AddThoughtImages.this, "Error: Image is not available!", Toast.LENGTH_LONG).show();
        }
        else
        {
            StoreImageInformation();
        }
    }
    private void StoreImageInformation()
    {
        loadingBar.setTitle("Add New Image");
    }
}

```

```

        loadingBar.setMessage("Dear User, please wait while we are adding the new
Image.");
        loadingBar.setCanceledOnTouchOutside(false);
        loadingBar.show();

        final StorageReference filePath =
ThoughtImagesRef.child(added_by).child(ImageUri.getLastPathSegment() + "-"
+ImageRandomKey + ".jpg");

        final UploadTask uploadTask = filePath.putFile(ImageUri);

        uploadTask.addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e)
            {
                String message = e.toString();
                Toast.makeText(AddThoughtImages.this, "Error: " + message,
Toast.LENGTH_SHORT).show();
                loadingBar.dismiss();
            }
        }).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot)
            {
                Toast.makeText(AddThoughtImages.this, "Product Image uploaded
Successfully...", Toast.LENGTH_SHORT).show();

                Task<Uri> urlTask = uploadTask.continueWithTask(new
Continuation<UploadTask.TaskSnapshot, Task<Uri>>() {
                    @Override
                    public Task<Uri> then(@NonNull Task<UploadTask.TaskSnapshot>
task) throws Exception
                    {
                        if (!task.isSuccessful())
                        {
                            throw task.getException();
                        }

                        downloadImageUrl = filePath.getDownloadUrl().toString();
                        return filePath.getDownloadUrl();
                    }
                }).addOnCompleteListener(new OnCompleteListener<Uri>() {
                    @Override
                    public void onComplete(@NonNull Task<Uri> task)
                    {
                        if (task.isSuccessful())
                        {
                            downloadImageUrl = task.getResult().toString();

                            Toast.makeText(AddThoughtImages.this, "got the Product
image Url Successfully...", Toast.LENGTH_SHORT).show();

                            SaveThoughtImagesToDatabase();
                        }
                    }
                });
            }
        });
    }
}

```

```

        });
    }
    private void SaveThoughtImagesToDatabase()
    {
        HashMap<String, Object> bookMap = new HashMap<>();
        bookMap.put("added_on", added_on);
        bookMap.put("added_by", added_by);
        bookMap.put("image", downloadImageUrl);

        ThoughtRef.child(ImageRandomKey).updateChildren(bookMap)
            .addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task)
                {
                    if (task.isSuccessful())
                    {
                        Intent intent = new Intent(AddThoughtImages.this,
ThoughtImagesActivity.class);
                        startActivity(intent);

                        loadingBar.dismiss();
                        Toast.makeText(AddThoughtImages.this, "Image is added
successfully..", Toast.LENGTH_SHORT).show();
                    }
                    else
                    {
                        loadingBar.dismiss();
                        String message = task.getException().toString();
                        Toast.makeText(AddThoughtImages.this, "Error: " +
message, Toast.LENGTH_SHORT).show();
                    }
                }
            });
    }
    private void OpenGallery()
    {
        Intent galleryIntent = new Intent();
        galleryIntent.setAction(Intent.ACTION_GET_CONTENT);
        galleryIntent.setType("image/*");
        startActivityForResult(galleryIntent, GalleryPick);
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data)
    {
        super.onActivityResult(requestCode, resultCode, data);

        if (requestCode==GalleryPick && resultCode==RESULT_OK && data!=null)
        {
            ImageUri = data.getData();
            thoughtImageView.setImageURI(ImageUri);
        }
    }
    private void init() {

        current_user= FirebaseAuth.getInstance().getCurrentUser();
        assert current_user != null;
        added_by=current_user.getUid();
        Log.v("LOG", added_by);
        ThoughtImagesRef =

```

```

        FirebaseStorage.getInstance().getReference().child("Thought of the Day Images");
        ThoughtRef = FirebaseDatabase.getInstance().getReference().child("Thought
of the Day");
        thoughtImageView = (ImageView) findViewById(R.id.uploadImage);
        UploadBtn = (Button) findViewById(R.id.btnUploadImage);
        calendar = Calendar.getInstance();
        loadingBar=new ProgressDialog(this);
        SimpleDateFormat currentDate = new SimpleDateFormat("MMM dd, yyyy");
        saveCurrentDate = currentDate.format(calendar.getTime());
        SimpleDateFormat currentTime = new SimpleDateFormat("HH:mm:ss a");
        savecurrentTime = currentTime.format(calendar.getTime());
        added_on=saveCurrentDate+" "+savecurrentTime;
        ImageRandomKey = ThoughtRef.push().getKey();
    }
}

```

AdminLoginActivity

```

package com.squares.adminapp;

import android.app.AlertDialog;
import android.content.Intent;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

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

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.adminapp.Admin.User;
import com.squares.adminapp.Models.AdminModel;
import com.squares.adminapp.Prevalent.Prevalent;
//import com.squares.adminapp.User.User;

import io.paperdb.Paper;

public class AdminLoginActivity extends AppCompatActivity {
    private FirebaseAuth mAuth;
    private ProgressDialog loadingBar;
    EditText EmailLogin, PasswordLogin;
    Button LoginButton;
    private TextView ForgetPassword;
    private String parentDbName = "Admins";
    private CheckBox chkBoxRememberMe;

```

```

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

    mAuth = FirebaseAuth.getInstance();

    init();
    listeners();
}

private void init() {
    EmailLogin = (EditText) findViewById(R.id.Login_email_input);
    PasswordLogin = (EditText) findViewById(R.id.Login_password_input);
    loadingBar = new ProgressDialog(this);
    LoginButton = (Button) findViewById(R.id.login_btn);
    chkBoxRememberMe = (CheckBox) findViewById(R.id.remember_me_chkb);
    ForgetPassword = (TextView) findViewById(R.id.forget_password_link);
    Paper.init(this);
}

private void listeners() {
    LoginButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            AllowUserToLogin();
        }
    });
    ForgetPassword.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent forgetPasswordIntent = new Intent(AdminLoginActivity.this,
ForgetPasswordAdminActivity.class);
            startActivity(forgetPasswordIntent);
        }
    });
}

private ValueEventListener getLoggedInUserFromDb(FirebaseAuth fbAuth) {
    return new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            // Get Post object and use the values to update the UI

            //AdminModel user= dataSnapshot.getValue(AdminModel.class);
            // User user = dataSnapshot.getValue(User.class);
            User user = dataSnapshot.getValue(User.class);

            if(!user.email.equals(fbAuth.getCurrentUser().getEmail())) {
                fbAuth.signOut();
                Toast.makeText(AdminLoginActivity.this, "Error: You doesn't have
access", Toast.LENGTH_SHORT).show();
                loadingBar.dismiss();
                return;
            }
            SendUserToMainActivity();
            Toast.makeText(AdminLoginActivity.this, "Logged in Successfully",
Toast.LENGTH_SHORT).show();
        }
    };
}

```

```

        loadingBar.dismiss();
        // ...
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {
        // Getting Post failed, log a message
        fbAuth.signOut();
        Toast.makeText(AdminLoginActivity.this, "Error: You doesn't have
access", Toast.LENGTH_SHORT).show();
        loadingBar.dismiss();
        // ...
    }
};

}

private void AllowUserToLogin() {
    String email = EmailLogin.getText().toString().trim();
    String password = PasswordLogin.getText().toString().trim();
    if (email.isEmpty()) {
        EmailLogin.setError("Email is required");
        EmailLogin.requestFocus();
        return;
    }
    if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()) {
        EmailLogin.setError("Please provide valid email");
        EmailLogin.requestFocus();
        return;
    }
    if (password.isEmpty()) {
        PasswordLogin.setError("Password is required");
        PasswordLogin.requestFocus();
        return;
    } else {

        if(chkBoxRememberMe.isChecked())
        {
            Paper.book().write(Prevalent.UserEmailKey, email);
            Paper.book().write(Prevalent.UserPasswordKey, password);
        }

        loadingBar.setTitle("Login");
        loadingBar.setMessage("Please wait..");
        loadingBar.setCanceledOnTouchOutside(true);
        loadingBar.show();

        mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {
                        FirebaseAuth fbAuth = FirebaseAuth.getInstance();
                        FirebaseUser user = fbAuth.getCurrentUser();
                        assert user != null;
                        if (user.isEmailVerified()) {
                            DatabaseReference userDbRef =
FireatabaseDatabase.getInstance().getReference("Admins").child(user.getUid());
                            userDbRef.addListenerForSingleValueEvent(getLoggedInUserFromDb(fbAuth));
                        }
                    }
                }
            });
    }
}

```

```

        } else {
            user.sendEmailVerification();
            Toast.makeText(AdminLoginActivity.this, "Check
Your Email to verify it", Toast.LENGTH_SHORT).show();
            loadingBar.dismiss();
        }

    } else {
        String message = task.getException().toString();
        SendUserToRegisterActivity();
        Toast.makeText(AdminLoginActivity.this, "Error:" +
message, Toast.LENGTH_SHORT).show();
        loadingBar.dismiss();
    }
}

private void SendUserToRegisterActivity() {
    Intent registerIntent = new Intent(AdminLoginActivity.this,
AdminLoginActivity.class);
    startActivity(registerIntent);
}

private void SendUserToMainActivity() {
    Intent MainIntent = new Intent(AdminLoginActivity.this,
HomeAdminActivity.class);
    startActivity(MainIntent);
}

}

```

AddMainActivity

```

package com.squares.adminapp;

import androidx.appcompat.app.AppCompatActivity;

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

public class AdminMainActivity extends AppCompatActivity {

    Button ContinueButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_admin_main);
        init();
        listeners();
    }

    private void listeners() {
        ContinueButton.setOnClickListener(new View.OnClickListener() {

```

```

        @Override
        public void onClick(View view) {
            Intent intent=new
Intent(AdminMainActivity.this,AdminLoginActivity.class);
            startActivity(intent);
        }
    });

    private void init() {
        ContinueButton=(Button) findViewById(R.id.continueBtn);
    }
}

```

AdminUpdateProfileActivity

```

package com.squares.adminapp;

import android.app.AlertDialog;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

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

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.google.firebaseio.storage.FirebaseStorage;
import com.google.firebaseio.storage.StorageReference;
import com.google.firebaseio.storage.UploadTask;
import com.squares.adminapp.Admin.User;
import com.squares.adminapp.Models.AdminModel;

import org.jetbrains.annotations.NotNull;

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

public class AdminUpdateProfile extends AppCompatActivity {

```

```

private EditText newUserName;
private String FullName;
private TextView newUserEmail, changepassword;
private Button save;
private String currentUserID, CurrentUserFullName, CurrentUserEmail;
private FirebaseAuth user;
private ProgressDialog loadingBar;
private DatabaseReference reference;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_my_profile);
    Toolbar toolbar = findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    init();
    RetrieveData();
    listeners();
}

private void listeners() {
    save.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            UpdateInformation();
        }
    });
    changepassword.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent forgetPasswordIntent = new Intent(AdminUpdateProfile.this,
ForgetPasswordAdminActivity.class);
            startActivity(forgetPasswordIntent);
        }
    });
}

private void UpdateInformation() {
    loadingBar.setTitle("Updating Profile");
    loadingBar.setMessage("Dear User, please wait while we are updating.");
    loadingBar.setCanceledOnTouchOutside(false);
    loadingBar.show();
    FullName = newUserName.getText().toString();

    if(FullName.isEmpty())
    {
        newUserName.setError("Full name is required");
        newUserName.requestFocus();
    }

    HashMap<String, Object> UserDataMap = new HashMap<>();
    UserDataMap.put("fullName", FullName);
    FirebaseDatabase.getInstance().getReference("Admins")
        .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
        .updateChildren(UserDataMap)
        .addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task)
            {

```

```

        if (task.isSuccessful())
        {
            loadingBar.dismiss();
            Toast.makeText(AdminUpdateProfile.this, "Profile Updated..", Toast.LENGTH_SHORT).show();
        }
        else
        {
            loadingBar.dismiss();
            String message = task.getException().toString();
            Toast.makeText(AdminUpdateProfile.this, "Error: " + message, Toast.LENGTH_SHORT).show();
        }
    });

private void RetrieveData() {
    user= FirebaseAuth.getInstance().getCurrentUser();
    reference= FirebaseDatabase.getInstance().getReference("Admins");
    currentUserID=user.getUid();
    reference.child(currentUserID);
    reference.child(currentUserID).addSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {

            User userProfile = snapshot.getValue(User.class);
            if (userProfile != null) {
                CurrentUserFullName = userProfile.fullName;
                newUserName.setText(CurrentUserFullName);

                CurrentUserEmail = userProfile.email;
                newUserEmail.setText(CurrentUserEmail);

            }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(AdminUpdateProfile.this, "Something Wrong!", Toast.LENGTH_SHORT).show();
        }
    });
}

private void init() {
    newUserName = findViewById(R.id.edit_text_name);
    newUserEmail = findViewById(R.id.text_email);
    save = findViewById(R.id.button_save);
    changepassword=findViewById(R.id.text_password);
    loadingBar=new ProgressDialog(this);
}
}

```

AdminEducationalInstituteActivity

```
package com.squares.adminapp;

import android.os.Bundle;

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

import com.google.android.material.tabs.TabLayout;

public class AdminEducationalInstitutesActivity extends AppCompatActivity {
    private Toolbar mToolbar;
    private ViewPager myEIVViewPager;
    private TabLayout myEITabLayout;
    private TabAccessorAdapterForEI myEITabsAccessorAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_admin_educational_institutes);
        myEIVViewPager=findViewById(R.id.my_main_tabs_pager);
        myEITabsAccessorAdapter= new
        TabAccessorAdapterForEI(getSupportFragmentManager());
        myEIVViewPager.setAdapter(myEITabsAccessorAdapter);
        myEITabLayout=(TabLayout) findViewById(R.id.my_main_tabs);
        myEITabLayout.setupWithViewPager(myEIVViewPager);

    }
}
```

BookPostRequest

```
package com.squares.adminapp;

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

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

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.adminapp.Adapters.BooksPostAdapter;
import com.squares.adminapp.Adapters.CategoryAdapter;
import com.squares.adminapp.Models.BooksPostModel;
import com.squares.adminapp.Models.CategoryModel;

public class BooksPostsRequest extends AppCompatActivity {

    RecyclerView postsRecyclerView;
    Button ViewPost;
```

```

BooksPostAdapter adapter;

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

    this.init();
    listeners();

    postsRecyclerView.setLayoutManager(new LinearLayoutManager(this));

    FirebaseRecyclerOptions<BooksPostModel> options =
        new FirebaseRecyclerOptions.Builder<BooksPostModel>()

    .setQuery(FirebaseDatabase.getInstance().getReference("Books").orderByChild("status")
    .equalTo("PENDING"), BooksPostModel.class)
    .build();

    adapter=new BooksPostAdapter(options);
    postsRecyclerView.setAdapter(adapter);

}

private void listeners() {
    ViewPost.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent=new
Intent(BooksPostsRequest.this,ViewFullBookPost.class);
            startActivity(intent);
        }
    });
}

private void init() {
    postsRecyclerView=(RecyclerView) findViewById(R.id.recycleViewPosts);
    ViewPost=(Button)findViewById(R.id.post_view_button);
}

@Override
protected void onStart() {
    super.onStart();
    adapter.startListening();
}

@Override
protected void onStop() {
    super.onStop();
    adapter.stopListening();
}
}

```

BookStatusActivity

```
package com.squares.adminapp;
```

```

import android.os.Bundle;

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

import com.google.android.material.tabs.TabLayout;

public class BookStatusActivity extends AppCompatActivity {
    private Toolbar mToolbar;
    private ViewPager myViewPager;
    private TabLayout myTabLayout;
    private TabsAccessorAdapter myTabsAccessorAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_book_status);
        myViewPager=findViewById(R.id.main_tabs_pager);
        myTabsAccessorAdapter=new
        TabsAccessorAdapter(getSupportFragmentManager());
        myViewPager.setAdapter(myTabsAccessorAdapter);

        myTabLayout=(TabLayout) findViewById(R.id.main_tabs);
        myTabLayout.setupWithViewPager(myViewPager);
    }
}

```

PublicInstituteFragment

```

package com.squares.adminapp;

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

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.adminapp.Adapters.EducationalInstituteAdapter;
import com.squares.adminapp.Models.EducationalInstituteModel;

/**
 * A simple {@link Fragment} subclass.
 * Use the {@link CollegeFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class CollegeFragment extends Fragment {
    RecyclerView postsRecyclclerView;
    EducationalInstituteAdapter adapter;

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

```

```

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

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

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

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (getArguments() != null) {
        mParam1 = getArguments().getString(ARG_PARAM1);
        mParam2 = getArguments().getString(ARG_PARAM2);
    }
}

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                        Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    View root=inflater.inflate(R.layout.fragment_college, container, false);

    postsRecyclcerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

    postsRecyclcerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

    postsRecyclcerView.setLayoutManager(new
LinearLayoutManager(getContext()));

    FirebaseRecyclerOptions<EducationalInstituteModel> options =
        new FirebaseRecyclerOptions.Builder<EducationalInstituteModel>()

.setQuery(FirebaseDatabase.getInstance().getReference("EducationalInstitute").orde
rByChild("Type").equalTo("Public"), EducationalInstituteModel.class)
        .build();
}

```

```

        adapter=new EducationalInstituteAdapter(options);
        postsRecyclerview.setAdapter(adapter);

    }

    return root;
}

@Override
public void onStart() {
    super.onStart();
    adapter.startListening();
}

@Override
public void onStop() {
    super.onStop();
    adapter.stopListening();
}
}
}

```

ForgetPasswordAdminActivity

```

package com.squares.adminapp;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

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

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

public class ForgetPasswordAdminActivity extends AppCompatActivity {

    EditText Email;
    Button Reset_password;
    private FirebaseAuth mAuth;
    private ProgressDialog loadingBar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forget_password_admin);
        mAuth = FirebaseAuth.getInstance();
        init();
        listeners();
    }

    private void listeners() {
        Reset_password.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

```

```

        ResetPassword();
    }
}
}

private void ResetPassword() {
    String email = Email.getText().toString().trim();
    if (email.isEmpty()) {
        Email.setError("Email is required");
        Email.requestFocus();
        return;
    }
    if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()) {
        Email.setError("Please provide valid email");
        Email.requestFocus();
        return;
    } else {
        loadingBar.setTitle("Sign in");
        loadingBar.setMessage("Please wait..");
        loadingBar.setCanceledOnTouchOutside(true);
        loadingBar.show();

        mAuth.sendPasswordResetEmail(email)
            .addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    if (task.isSuccessful()) {
                        Toast.makeText(ForgetPasswordAdminActivity.this,
                        "Check Your Email to Reset your password", Toast.LENGTH_SHORT).show();
                        loadingBar.dismiss();
                    } else {
                        Toast.makeText(ForgetPasswordAdminActivity.this,
                        "Try Again! Something wrong happened", Toast.LENGTH_SHORT).show();
                        loadingBar.dismiss();
                    }
                }
            });
    }
}

private void init() {
    Email = (EditText) findViewById(R.id.reset_password_email_input);
    Reset_password = (Button) findViewById(R.id.reset_password_btn);
    loadingBar = new ProgressDialog(this);
}
}

```

HomeAdminActivity

```

package com.squares.adminapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

```

```

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.cardview.widget.CardView;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.adminapp.Admin.User;
import com.squares.adminapp.Models.AdminModel;

import java.text.SimpleDateFormat;
import java.util.Calendar;

public class HomeAdminActivity extends AppCompatActivity {

    CardView
MyAccountCv,AddBookCategoryCv,AddThoughtCv,EducatInstCv,BookPostRequestCv,RecommendedDataCv,UsersCv,CsvCv;
    // Toolbar toolbar;
    MenuItem Logout,Settings;
    private FirebaseAuth user;
    private DatabaseReference reference;
    TextView dateDateTimeTv,currentUserId;
    String currentUserID,CurrentUserFullName;
    private String currentDate,currentTime;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home_admin);
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        init();
        listeners();
        showCurrentDateTime();
        showCurrentUser();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.logout) {
            FirebaseAuth.getInstance().signOut();
            startActivity(new

```

```

Intent(HomeAdminActivity.this, AdminLoginActivity.class));
        return true;
    }

    if (id == R.id.settings) {
        Toast.makeText(this, "Setting", Toast.LENGTH_LONG).show();
        return true;
    }

    return super.onOptionsItemSelected(item);
}

private void showCurrentUser() {
    user= FirebaseAuth.getInstance().getCurrentUser();
    reference= FirebaseDatabase.getInstance().getReference("Admins");
    currentUserID=user.getUid();
    reference.child(currentUserID).addListenerForSingleValueEvent(new
ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {

            User userProfile = snapshot.getValue(User.class);
            if(userProfile!=null)
            {
                CurrentUserFullName=userProfile.fullName;

                currentUserTv.setText("Welcome "+CurrentUserFullName+" !");

            }
        }

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(HomeAdminActivity.this,"Something
Wrong!",Toast.LENGTH_SHORT).show();
        }
    });
}

private void showCurrentDateTime() {
    Calendar calendar = Calendar.getInstance();

    SimpleDateFormat _currentDate = new SimpleDateFormat("MMM dd, yyyy");
    currentDate = _currentDate.format(calendar.getTime());

    SimpleDateFormat _currentTime = new SimpleDateFormat("HH:mm:ss a");
    currentTime = _currentTime.format(calendar.getTime());

    dateTimeTv.setText(currentDate+" "+currentTime);
}

private void listeners() {
    MyAccountCv.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent=new
Intent(HomeAdminActivity.this,AdminUpdateProfile.class);

```

```

        startActivity(intent);
    }
});
AddBookCategoryCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeAdminActivity.this,ShowBookCategoriesActivity.class);
        startActivity(intent);
    }
});
AddThoughtCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeAdminActivity.this,ThoughtImagesActivity.class);
        startActivity(intent);
    }
});
BookPostRequestCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeAdminActivity.this,BookStatusActivity.class);
        startActivity(intent);
    }
});
UsersCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeAdminActivity.this,ViewUsersActivity.class);
        startActivity(intent);
    }
});
EducatInstCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeAdminActivity.this,AminEducationalInstitutesActivity.class);
        startActivity(intent);
    }
});
}

private void init() {
    dateTimeTv=(TextView)findViewById(R.id.dateTime);
    currentUserTv=(TextView)findViewById(R.id.textWelcomeUser);
    MyAccountCv=(CardView)findViewById(R.id.MyAccount);
    AddBookCategoryCv=(CardView)findViewById(R.id.addBookCategory);
    AddThoughtCv=(CardView)findViewById(R.id.Addthoughts);
    EducatInstCv=(CardView)findViewById(R.id.EducationalInstitutesCrawler);
    BookPostRequestCv=(CardView)findViewById(R.id.BookRequest);
    RecommendedDataCv=(CardView)findViewById(R.id.RecommendedData);
    UsersCv=(CardView)findViewById(R.id.ViewUsersCardView);
    CsvCv=(CardView)findViewById(R.id.Csvdata);
    Logout=(MenuItem)findViewById(R.id.logout);
    Settings=(MenuItem)findViewById(R.id.settings);
}

```

```
}
```

PendingPostOfBooksFragment

```
package com.squares.adminapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.FirebaseDatabase;
import com.squares.adminapp.Adapters.BooksPostAdapter;
import com.squares.adminapp.Models.BooksPostModel;

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

    RecyclerView postsRecyclcerView;
    Button ViewPost;
    BooksPostAdapter adapter;

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

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

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

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

```

        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View root=inflater.inflate(R.layout.fragment_pending_post_of_books,
        container, false);

        postsRecyclerview=(RecyclerView)
        root.findViewById(R.id.recycleViewPosts);
        ViewPost=(Button)root.findViewById(R.id.post_view_button);

        postsRecyclerview.setLayoutManager(new
        LinearLayoutManager(getContext()));

        FirebaseRecyclerOptions<BooksPostModel> options =
            new FirebaseRecyclerOptions.Builder<BooksPostModel>()

        .setQuery(FirebaseDatabase.getInstance().getReference("Books").orderByChild("status")
        .equalTo("PENDING"), BooksPostModel.class)
        .build();

        adapter=new BooksPostAdapter(options);
        postsRecyclerview.setAdapter(adapter);

        ViewPost.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(getActivity(),ViewFullBookPost.class);
                startActivity(intent);
            }
        });

        return root;
    }

    //    @Override
    //    public void onClick(View v) {
    //        // implements your things
    //        Intent intent=new Intent(getActivity(),ViewFullBookPost.class);
    //        startActivity(intent);
    //    }
}

```

```

    @Override
    public void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    public void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

RejectedBookPostFragment

```

package com.squares.adminapp;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.FirebaseDatabase;
import com.squares.adminapp.Adapters.BooksPostAdapter;
import com.squares.adminapp.Models.BooksPostModel;

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

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

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

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

    /**
     * Use this factory method to create a new instance of
     * this fragment using the provided parameters.
     *
     * @param param1 Parameter 1.
     */

```

```

    * @param param2 Parameter 2.
    * @return A new instance of fragment RejectedBookPostsFragment.
    */
    // TODO: Rename and change types and number of parameters
    public static RejectedBookPostsFragment newInstance(String param1, String param2)
    {
        RejectedBookPostsFragment fragment = new RejectedBookPostsFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View root=inflater.inflate(R.layout.fragment_rejected_book_posts,
        container, false);

        postsRecyclcerView=(RecyclerView)
        root.findViewById(R.id.recycleViewPosts);
        ViewPost=(Button)root.findViewById(R.id.post_view_button);

        postsRecyclcerView.setLayoutManager(new
        LinearLayoutManager(getContext()));

        FirebaseRecyclerOptions<BooksPostModel> options =
            new FirebaseRecyclerOptions.Builder<BooksPostModel>()

        .setQuery(FirebaseDatabase.getInstance().getReference("Books").orderByChild("status")
        .equalTo("DECLINED"), BooksPostModel.class)
            .build();

        adapter=new BooksPostAdapter(options);
        postsRecyclcerView.setAdapter(adapter);

        // ViewPost.setOnClickListener(new View.OnClickListener() {
        //     @Override
        //     public void onClick(View view) {
        //         Intent intent=new Intent(getActivity(),ViewFullBookPost.class);
        //         startActivity(intent);
        //     }
        // });

        return root;
    }
    @Override

```

```

    public void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    public void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

ShowBookCategoriesActivity

```

RecyclerView ShowExistingCategoriesRecycV;
Button AddNewCategory;
CategoryAdapter adapter;

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

    init();
    listeners();

    ShowExistingCategoriesRecycV.setLayoutManager(new LinearLayoutManager(this));

    FirebaseRecyclerOptions<CategoryModel> options =
        new FirebaseRecyclerOptions.Builder<CategoryModel>()

    .setQuery(FirebaseDatabase.getInstance().getReference("bookCategories"),
CategoryModel.class)
        .build();

    adapter=new CategoryAdapter(options);
    ShowExistingCategoriesRecycV.setAdapter(adapter);
}

private void listeners() {
    AddNewCategory.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent=new
Intent(ShowBookCategoriesActivity.this,AddBookCategoryActivity.class);
            startActivity(intent);
        }
    });
}

private void init() {
    AddNewCategory=(Button)findViewById(R.id.addNewBookCategory);
    ShowExistingCategoriesRecycV=(RecyclerView)
findViewById(R.id.recycleViewECategories);

}

@Override
protected void onStart() {
    super.onStart();
}

```

```
        adapter.startListening();
    }

@Override
protected void onStop() {
    super.onStop();
    adapter.stopListening();
}
```

TabAccessorAdapterForEI

```
package com.squares.adminapp;

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

import com.google.firebaseio.database.annotations.Nullable;

public class TabAccessorAdapterForEI extends FragmentPagerAdapter {
    public TabAccessorAdapterForEI(@NonNull FragmentManager fm) {
        super(fm);
    }

    @NonNull
    @Override
    public Fragment getItem(int position) {
        switch (position) {
            case 0:
                CollegeFragment collegeFragment = new CollegeFragment();
                return collegeFragment;
            case 1:
                UniversityFragment universityFragment = new UniversityFragment();
                return universityFragment;
            default:
                return null;
        }
    }

    @Override
    public int getCount() {
        return 2;
    }

    @Nullable
    @Override
    public CharSequence getPageTitle(int position) {

        switch (position) {
            case 0:
                return "Public";
            case 1:
                return "Private";
            default:
                return null;
        }
    }
}
```

TabAccessorAdapter

```

package com.squares.adminapp;

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

import com.google.firebase.database.annotations.Nullable;

public class TabsAccessorAdapter extends FragmentPagerAdapter {
    public TabsAccessorAdapter(@NonNull FragmentManager fm) {
        super(fm);
    }

    @NonNull
    @Override
    public Fragment getItem(int position) {
        switch (position) {
            case 0:
                PendingPostOfBooksFragment pendingPostOfBooksFragment = new
PendingPostOfBooksFragment();
                return pendingPostOfBooksFragment;
            case 1:
                AcceptedBookPostsFragment acceptedBookPostsFragment = new
AcceptedBookPostsFragment();
                return acceptedBookPostsFragment;
            case 2:
                RejectedBookPostsFragment rejectedBookPostsFragment = new
RejectedBookPostsFragment();
                return rejectedBookPostsFragment;
            default:
                return null;
        }
    }

    @Override
    public int getCount() {
        return 3;
    }

    @Nullable
    @Override
    public CharSequence getPageTitle(int position) {

        switch (position) {
            case 0:
                return "Pending";
            case 1:
                return "Accepted";
            case 2:
                return "Rejected";
            default:
                return null;
        }
    }
}

```

ThoughtImagesActivity

```

package com.squares.adminapp;

import android.content.Intent;

```

```

import android.os.Bundle;
import android.view.View;
import android.widget.Button;

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

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.adminapp.Adapters.ThoughtImagesAdapter;
import com.squares.adminapp.Models.ThoughtModel;

import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;

public class ThoughtImagesActivity extends AppCompatActivity {

    Button AddNewThought;
    RecyclerView ShowThoughtImages;

    ThoughtImagesAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_thought_of_the_day);
        init();
        listeners();
        ShowThoughtImages.setLayoutManager(new LinearLayoutManager(this));
        FirebaseRecyclerOptions<ThoughtModel> options =
            new FirebaseRecyclerOptions.Builder<ThoughtModel>()

        .setQuery(FirebaseDatabase.getInstance().getReference("Thought of the Day"),
        ThoughtModel.class)
            .build();

        adapter=new ThoughtImagesAdapter(options);
        ShowThoughtImages.setAdapter(adapter);
    }

    private void listeners() {
        AddNewThought.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new
                Intent(ThoughtImagesActivity.this,AddThoughtImages.class);
                startActivity(intent);
            }
        });
    }

    private void init() {

```

```

        AddNewThought=(Button)findViewById(R.id.addNewThought);
        ShowThoughtImages=(RecyclerView)
findViewById(R.id.recycleViewThoughtImages);
    }

    @Override
    protected void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    protected void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

PrivateInstituteFragment

```

package com.squares.adminapp;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.FirebaseDatabase;
import com.squares.adminapp.Adapters.BooksPostAdapter;
import com.squares.adminapp.Adapters.EducationalInstituteAdapter;
import com.squares.adminapp.Models.BooksPostModel;
import com.squares.adminapp.Models.EducationalInstituteModel;

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

    RecyclerView postsRecyclcerView;
    EducationalInstituteAdapter adapter;

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

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

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

```

```

    }

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

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View root=inflater.inflate(R.layout.fragment_university, container, false);

        postsRecyclerview=(RecyclerView)
        root.findViewById(R.id.recycleViewPosts);

        postsRecyclerview.setLayoutManager(new
        LinearLayoutManager(getContext()));

        FirebaseRecyclerOptions<EducationalInstituteModel> options =
            new FirebaseRecyclerOptions.Builder<EducationalInstituteModel>()

        .setQuery(FirebaseDatabase.getInstance().getReference("EducationalInstitute").orde
        rByChild("Type").equalTo("Private"), EducationalInstituteModel.class)
            .build();

        adapter=new EducationalInstituteAdapter(options);
        postsRecyclerview.setAdapter(adapter);

        return root;
    }

    @Override
    public void onStart() {

```

```

        super.onStart();
        adapter.startListening();
    }

    @Override
    public void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

ADMIN LAYOUTS CODE

Activity_add_book-category

```

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textAlignment="center"
        android:text="Add Book Category"
        android:gravity="center_horizontal"
        android:layout_margin="15dp"
        android:padding="8dp"
        android:textSize="20sp"
        />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"
        android:orientation="horizontal"/>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Enter New Book Category name"
        android:padding="8dp"
        android:textSize="14sp"
        android:textColor="@color/colorAccent"
        android:layout_margin="8dp"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"
        android:orientation="horizontal"/>
    <EditText

```

```

        android:id="@+id/addNewCatEt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter New Book Category name"
        android:padding="8dp"
        android:background="#0000"
        android:textSize="14sp"
        android:inputType="text"
        android:textColor="@color/colorAccent"
        android:layout_marginRight="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginBottom="8dp"
    />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"
    android:orientation="horizontal"/>

<Button
    android:id="@+id/AddNewCategoryBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:padding="8dp"
    android:layout_margin="8dp"
    android:text="Add Category"/>

</LinearLayout>

```

Activity_book_posted_for_approval

```

<?xml version="1.0" encoding="utf-8"?>
<GridLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:ignore="MissingDefaultResource">
    <LinearLayout
        android:padding="3dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/user_request_book_post_image"
            android:layout_width="83dp"
            android:layout_height="89dp" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">
                <TextView
                    android:id="@+id/user_request_book_posted_by"
                    android:layout_width="200dp"

```

```
        android:layout_height="wrap_content"
        android:text="Posted by"
        android:layout_margin="5dp"
        android:textAlignment="textStart"
        android:textSize="20sp"
        android:textStyle="italic"
        android:gravity="start">
    </TextView>
</LinearLayout>

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

    <Button
        android:id="@+id/post_view_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="2dp"
        android:background="@drawable/input_design"
        android:text="View"
        android:textColor="@color/colorAccent"
        android:textSize="12sp"
        android:textStyle="italic">

    </Button>

    <Button
        android:id="@+id/post_approve_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="2dp"
        android:background="@drawable/input_design"
        android:text="Approve"
        android:textColor="@color/colorAccent"
        android:textSize="12sp"
        android:textStyle="italic">

    </Button>

    <Button
        android:id="@+id/post_cancel_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="2dp"
        android:background="@drawable/input_design"
        android:text="Reject"
        android:textColor="@color/colorAccent"
        android:textSize="12sp"
        android:textStyle="italic">

    </Button>
</LinearLayout>
</LinearLayout>
</LinearLayout>

</GridLayout>
```

Activity_admin-login

```
<?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"
    android:background="@drawable/loginbg"
    tools:context=".AdminLoginActivity">

    <ImageView
        android:id="@+id/login_applogo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:src="@drawable/applogo"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="120dp"
        />
    <TextView
        android:id="@+id/admintitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/login_applogo"
        android:layout_centerInParent="true"
        android:text="ADMIN PANEL"
        android:textColor="@color/white"
        android:textSize="25sp"
        android:layout_marginBottom="8dp"/>

    <EditText
        android:id="@+id/login_email_input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/admintitle"
        android:background="@drawable/input_design_login"
        android:padding="16dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:hint="Email"
        android:inputType="textEmailAddress"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        android:textSize="16sp"
        />

    <EditText
        android:id="@+id/login_password_input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/login_email_input"
        android:background="@drawable/input_design_login"
        android:padding="16dp"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:layout_marginTop="6dp"
        android:hint="Password"
```

```
        android:inputType="textPassword"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        android:textSize="16sp"
    />

<LinearLayout
    android:id="@+id/linear_layout_1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_below="@+id/login_password_input"
    android:layout_marginTop="5dp"
>

<CheckBox
    android:id="@+id/remember_me_chkb"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

    android:text="Remember me"
    android:textColor="@color/white"
    android:visibility="invisible"
    android:gravity="center_vertical"
    android:textSize="14.5sp"
    android:layout_marginLeft="17dp"
/>

<TextView
    android:id="@+id/forget_password_link"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Forget Password?"
    android:textColor="@color/white"
    android:textSize="14.5sp"
    android:layout_marginLeft="70dp"
/>

</LinearLayout>

<Button
    android:id="@+id/login_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/linear_layout_1"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/buttons"
    android:padding="16dp"
    android:text="Login"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="15sp"
    app:backgroundTint="@color/colorPrimary" />
```

```

</RelativeLayout>
Activity_admin-main
<?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"
    android:background="@drawable/bghome"
    tools:context=".AdminMainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="WELCOME TO THE ADMIN PANEL"
        android:layout_centerInParent="true"
        android:textColor="@color/white"
        android:textSize="25dp"
        android:textAlignment="center"
        android:gravity="center_horizontal"
        android:textStyle="bold|italic"/>
    <Button
        android:id="@+id/continueBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:text="Continue"
        android:textColor="@color/colorPrimary"
        android:textStyle="bold|italic"
        app:backgroundTint="#FFFFFF" />

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

```

```

<ImageView
    android:layout_width="250dp"
    android:layout_height="250dp"
    android:layout_gravity="center"
    android:id="@+id/uploadImage"
    android:layout_margin="10dp"
    android:background="@drawable/ic_menu_camera"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"
    android:orientation="horizontal"/>
<Button
    android:text="Upload"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="13dp"
    android:id="@+id/btnUploadImage"/>

```

</LinearLayout>

Activity_admin_educational_institutes

```

<?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=".AminEducationalInstitutesActivity">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        tools:ignore="MissingConstraints">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
        <com.google.android.material.tabs.TabLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/my_main_tabs"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">
            </com.google.android.material.tabs.TabLayout>
    </com.google.android.material.appbar.AppBarLayout>
    <androidx.viewpager.widget.ViewPager
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/my_main_tabs_pager"
        android:layout_below="@+id/appBarLayout"
        tools:ignore="MissingConstraints">
        </androidx.viewpager.widget.ViewPager>
</androidx.constraintlayout.widget.ConstraintLayout>

```

Activity_book_status

```

<?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"
    android:background="@color/white"
    tools:context=".BookStatusActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">

        </include>
        <com.google.android.material.tabs.TabLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/main_tabs"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">

        </com.google.android.material.tabs.TabLayout>

    </com.google.android.material.appbar.AppBarLayout>
    <androidx.viewpager.widget.ViewPager
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/main_tabs_pager"
        android:layout_below="@+id/appBarLayout">

    </androidx.viewpager.widget.ViewPager>

</RelativeLayout>

```

Activity_book_post_request

```

<?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"
    tools:context=".BooksPostsRequest">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/my_appBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">


```

```
</include>
</com.google.android.material.appbar.AppBarLayout>

<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recycleViewPosts"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="8dp"/>
```

</LinearLayout>

Activity_educational_Institute

```
<?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"
    tools:context=".BooksPostsRequest">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewInstitutes"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>
```

</LinearLayout>

Activity_existing_categories

```
<?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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
```

```

</com.google.android.material.appbar.AppBarLayout>

<Button
    android:id="@+id/addNewBookCategory"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="right"
    android:layout_margin="10dp"
    android:padding="8dp"
    android:text="Add New Category"
    app:icon="@android:drawable/ic_input_add" />
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recycleViewCategories"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"/>

</LinearLayout>
```

Activity_forget_password

```

<?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"
    android:background="@drawable/loginbg">
    <ImageView
        android:id="@+id/app_logo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="90dp"
        android:src="@drawable/applogo"
        />
    <TextView
        android:id="@+id/loginPage"
        android:layout_marginTop="5dp"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:fontFamily="serif"
        android:layout_below="@+id/app_logo"
        android:gravity="center"
        android:text="Admin Panel "
        android:textAlignment="center"
        android:textColor="@color/white"
        android:textColorHighlight="@color/colorAccent"
        android:textSize="30sp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/loginPage2"
        android:layout_marginTop="5dp"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:fontFamily="serif"
        android:layout_below="@+id/loginPage"
        android:gravity="center"
```

```
        android:text="Change Password"
        android:textAlignment="center"
        android:textColor="@color/white"
        android:textColorHighlight="@color/colorAccent"
        android:textSize="20sp"
        android:textStyle="bold" />
<View
    android:layout_width="350dp"
    android:layout_height="1.5dp"
    android:background="@color/colorPrimaryDark"
    android:layout_below="@+id/loginPage2"
    android:layout_marginTop="5dp"
    android:layout_marginBottom="5dp"
    android:layout_centerInParent="true"/>

<EditText
    android:id="@+id/reset_password_email_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/loginPage2"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="25dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/input_design_login"
    android:drawableStart="@android:drawable/ic_dialog_email"
    android:fontFamily="serif"
    android:hint=" Email"
    android:inputType="textEmailAddress"
    android:padding="10dp"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
    android:drawableLeft="@android:drawable/ic_dialog_email" />

<Button
    android:id="@+id/reset_password_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/reset_password_email_input"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/buttons"
    android:padding="10dp"
    android:text="Reset password"
    android:textStyle="bold"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="15sp"
    app:backgroundTint="@color/colorPrimaryDark" />

</RelativeLayout>
```

Activity_home_admin

```
<?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="wrap_content">
    <LinearLayout
        android:orientation="vertical"
        android:background="@drawable/home_bg"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <com.google.android.material.appbar.AppBarLayout
            android:id="@+id/myAppBarLayout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">
            <include
                layout="@layout/toolbar"
                android:id="@+id/main_page_toolbar">
            </include>
        </com.google.android.material.appbar.AppBarLayout>
        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="196dp"
            android:layout_weight="1"
            android:padding="16dp">

            <TextView
                android:id="@+id/textWelcomeUser"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_above="@+id/dateTime"
                android:layout_alignParentEnd="true"
                android:layout_alignParentRight="true"
                android:layout_marginEnd="10dp"
                android:layout_marginRight="10dp"
                android:text="Welcome Admin"
                android:textColor="@android:color/white"
                android:textSize="20sp"
                android:textStyle="bold" />

            <TextView
                android:id="@+id/dateTime"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_alignParentEnd="true"
                android:layout_alignParentRight="true"
                android:layout_alignParentBottom="true"
                android:layout_marginEnd="10dp"
                android:layout_marginRight="10dp"
                android:layout_marginBottom="15dp"
                android:text="Tuesday, 6-ct-2020"
                android:textColor="@android:color/white"
                android:textSize="13sp"
                android:textStyle="bold|italic" />

        </RelativeLayout>
        <GridLayout
```

```
        android:id="@+id/mainGrid"
        android:layout_width="match_parent"
        android:layout_height="534dp"
        android:layout_gravity="bottom"
        android:layout_weight="6"
        android:alignmentMode="alignMargins"
        android:columnCount="2"
        android:columnOrderPreserved="false"
        android:padding="5dp"
        android:rowCount="4">

    <!-- Row 1 -->

    <!-- Column 1 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/MyAccount"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_rowWeight="1"
        android:layout_columnWeight="1"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginBottom="15dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="8dp">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|center_vertical"
            android:layout_margin="16dp"
            android:orientation="vertical">

            <ImageView
                android:id="@+id/accImg"
                android:layout_width="wrap_content"
                android:layout_height="49dp"
                android:layout_gravity="center_horizontal"
                android:src="@drawable/profile" />

            <TextView
                android:id="@+id/accID"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_horizontal"
                android:text="My Account"
                android:textAlignment="center"
                android:textColor="@android:color/black"
                android:textSize="15sp"
                android:textStyle="bold" />

        </LinearLayout>

    </androidx.cardview.widget.CardView>

    <!-- Column 2 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/addBookCategory"
        android:layout_width="0dp"
```

```
        android:layout_height="0dp"
        android:layout_rowWeight="1"
        android:layout_columnWeight="1"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="16dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="8dp" >

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical" >

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="49dp"
            android:layout_gravity="center_horizontal"
            android:src="@drawable/addbookcat" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:gravity="center_horizontal"
            android:text="Add Book Category"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>
<!-- Row 2 -->

<!-- Column 1 -->
<androidx.cardview.widget.CardView
    android:id="@+id/Addthoughts"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginLeft="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp" >

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical" >

        <ImageView
            android:layout_width="wrap_content"
```

```
        android:layout_height="49dp"
        android:layout_gravity="center_horizontal"
        android:src="@drawable/thoughts" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:gravity="center_horizontal"
        android:text="Add Thought of the Day"
        android:textAlignment="center"
        android:textColor="@android:color/black"
        android:textSize="15sp"
        android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>

<!-- Column 2 -->
<androidx.cardview.widget.CardView
    android:id="@+id/EducationalInstitutesCrawler"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginStart="5dp"
    android:layout_marginLeft="5dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="49dp"
            android:layout_gravity="center_horizontal"
            android:src="@drawable/educational_institutes" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:gravity="center_horizontal"
            android:text="Educational Institutes Data"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>

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

```
<!-- Row 2 -->
<androidx.cardview.widget.CardView
    android:id="@+id/BookRequest"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginLeft="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="49dp"
            android:layout_gravity="center_horizontal"
            android:src="@drawable/books" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:gravity="center_horizontal"
            android:text="Books Posts Request"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>

<!-- Column 2 -->
<androidx.cardview.widget.CardView
    android:id="@+id/RecommendedData"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginStart="5dp"
    android:layout_marginLeft="5dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="49dp"
        android:layout_gravity="center_horizontal"
        android:src="@drawable/recommended" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:gravity="center_horizontal"
        android:text="Recommended Data"
        android:textAlignment="center"
        android:textColor="@android:color/black"
        android:textSize="15sp"
        android:textStyle="bold" />

</LinearLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/ViewUsersCardView"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginLeft="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="49dp"
        android:layout_gravity="center_horizontal"
        android:src="@drawable/users" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:gravity="center_horizontal"
        android:text="View Users"
        android:textAlignment="center"
        android:textColor="@android:color/black"
        android:textSize="15sp"
        android:textStyle="bold" />

```

```

        </LinearLayout>

    </androidx.cardview.widget.CardView>

    <!-- Column 2 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/Csvdata"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_rowWeight="1"
        android:layout_columnWeight="1"
        android:layout_marginStart="5dp"
        android:layout_marginLeft="5dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="15dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="8dp">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|center_vertical"
            android:layout_margin="16dp"
            android:orientation="vertical">

            <ImageView
                android:layout_width="wrap_content"
                android:layout_height="49dp"
                android:layout_gravity="center_horizontal"
                android:src="@drawable/csv"/>

            <TextView
                android:layout_width="match_parent"
                android:layout_height="50dp"
                android:gravity="center_horizontal"
                android:text="CSV Data"
                android:textAlignment="center"
                android:textColor="@android:color/black"
                android:textSize="15sp"
                android:textStyle="bold" />

        </LinearLayout>

    </androidx.cardview.widget.CardView>

</GridLayout>

</LinearLayout>

</ScrollView>

```

Activity_my_profile

```

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

```

```
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <RelativeLayout
        android:layout_below="@+id/myAppBarLayout"
        android:id="@+id/layoutHead"
        android:background="@color/colorAccent"
        android:layout_width="match_parent"
        android:layout_height="200dp">

        <TextView
            android:textAppearance="@style/TextAppearance.AppCompat.Headline"
            android:textSize="18sp"
            android:textAlignment="center"
            android:layout_centerInParent="true"
            android:text="Enter Name"
            android:textColor="@color/white"
            android:background="#0000"
            android:textStyle="bold"
            android:id="@+id/tv"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal" />

        <EditText
            android:layout_below="@+id/tv"
            android:textAlignment="center"
            android:layout_marginTop="5dp"
            android:textColor="@android:color/white"
            android:background="#0000"
            android:layout_centerInParent="true"
            android:layout_centerHorizontal="true"
            android:enabled="true"
            android:textStyle="bold|italic"
            android:id="@+id/edit_text_name"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal" />

    </RelativeLayout>

    <LinearLayout
        android:layout_marginTop="10dp"
        android:padding="16dp"
        android:layout_centerHorizontal="true"
        android:orientation="vertical">
```

```

        android:layout_below="@+id/layoutHead"
        android:layout_width="360dp"
        android:layout_height="match_parent">

    <TextView
        android:textAppearance="@style/TextAppearance.AppCompat.Headline"
        android:layout_gravity="top"
        android:drawablePadding="16dp"
        android:textSize="18sp"
        android:textStyle="bold|italic"
        android:drawableLeft="@drawable/ic_email"
        android:background="#0000"
        android:id="@+id/text_email"
        android:layout_marginBottom="10dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <TextView
        android:textAppearance="@style/TextAppearance.AppCompat.Headline"
        android:layout_gravity="top"
        android:textSize="18sp"
        android:drawablePadding="16dp"
        android:drawableLeft="@drawable/ic_lock"
        android:text="Change Password"
        android:enabled="true"
        android:textStyle="bold|italic"
        android:id="@+id/text_password"
        android:layout_width="match_parent"
        android:layout_marginBottom="20dp"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/button_save"
        android:layout_marginTop="35dp"
        android:textColor="@android:color/white"
        android:background="?colorPrimary"
        android:textAllCaps="false"
        android:layout_gravity="center_horizontal"
        android:text="Save Changes"
        android:layout_width="140dp"
        android:layout_height="wrap_content"/>

    </LinearLayout>
</RelativeLayout>
</FrameLayout>

```

Activity_show_book_categories

```

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

```

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
<TextView
    android:id="@+id/catNameTv"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Category Name"
    android:padding="8dp"
    android:textSize="18sp"
    android:background="@color/white"
    android:textColor="@color/colorPrimaryDark"
    android:textStyle="bold|italic"
    android:layout_marginLeft="5dp"
    android:textAlignment="center"
    android:layout_margin="8dp"
    android:layout_row="0"
    android:layout_column="0"
    android:gravity="center_horizontal" />

</RelativeLayout>

<ImageView
    android:id="@+id/removeCat"
    android:layout_width="wrap_content"
    android:layout_height="50dp"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_gravity="right"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="8dp"
    android:backgroundTint="@color/colorPrimaryDark"
    android:padding="8dp"
    android:src="@android:drawable/ic_menu_delete" />

<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_marginTop="50dp"
    android:background="@color/colorPrimaryDark" />
</GridLayout>

```

Activity_show_thought_of_the_day

```

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

<androidx.cardview.widget.CardView
    android:id="@+id/thoughtCardView"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:cardCornerRadius="5dp"
        android:layout_margin="5dp">

    <ImageView
        android:id="@+id/thoughtImagesRetrieve"
        android:layout_width="match_parent"
        android:layout_height="250dp"
        android:scaleType="fitXY"
        android:layout_margin="10dp"
        android:background="@drawable/input_design"
        android:padding="5dp" />

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

```

Activity_show_users

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:ndroid="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"
    tools:ignore="MissingDefaultResource">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewShowUsers"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>

</LinearLayout>

```

Activity_thought_of_the_day

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:ndroid="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"
    tools:context=".ThoughtImagesActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <Button
        android:id="@+id/addNewThought"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:layout_margin="10dp"
        android:padding="8dp"
        android:text="Add New Image"
        android:icon="@android:drawable/ic_input_add" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewThoughtImages"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>

</LinearLayout>
```

Activity_users_cardView

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/thoughtCardView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="5dp"
    android:padding="16dp">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="15dp"
        android:layout_margin="8dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/profileImage"
            android:layout_width="80dp"
            android:layout_height="90dp"

            android:src="@mipmap/ic_launcher"
            app:civ_border_color="@color/colorPrimaryDark"
            app:civ_border_width="2dp" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/UserNameTxt"
            android:text="UserNameTxt"
            android:padding="3dp"
```

```
        android:textSize="18sp"
        android:layout_marginBottom="5dp"
        android:textColor="#000"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_marginLeft="10dp"/>
    <View
        android:layout_marginTop="5dp"
        android:layout_below="@+id/UserNameTxt"
        android:layout_width="match_parent"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_height="0.5dp"
        android:layout_marginBottom="5dp"
        android:background="@color/colorPrimaryDark" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/UserEmailTxt"
        android:text="UserEmailTxt"
        android:padding="3dp"
        android:textSize="18sp"
        android:layout_marginTop="5dp"
        android:textColor="#000"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_below="@+id/UserNameTxt"
        android:layout_marginLeft="10dp"/>
    <View
        android:layout_marginTop="5dp"
        android:layout_below="@+id/UserEmailTxt"
        android:layout_width="match_parent"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_height="0.5dp"
        android:layout_marginBottom="5dp"
        android:background="@color/colorPrimaryDark" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/UserAddedOnTv"
        android:text="UserAddedOnTv"
        android:textSize="18sp"
        android:padding="3dp"
        android:layout_marginTop="5dp"
        android:textColor="#000"
        android:layout_marginBottom="5dp"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_below="@+id/UserEmailTxt"
        android:layout_marginLeft="10dp"/>
    <Button
        android:id="@+id/ViewUser"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="View User"
        android:layout_toRightOf="@+id/profileImage"
        android:layout_below="@+id/UserAddedOnTv"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:layout_toEndOf="@+id/profileImage" />
```

```

<View
    android:layout_marginTop="5dp"
    android:layout_below="@+id/ViewUser"
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorPrimaryDark" />
</RelativeLayout>

</androidx.cardview.widget.CardView>

```

Activity_view_educational_institute

```

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

<RelativeLayout
    android:id="@+id/RL"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <androidx.cardview.widget.CardView
        android:layout_below="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        app:cardCornerRadius="10dp"
        android:layout_margin="4dp"
        android:clickable="true"
        android:id="@+id/cv_item"
        android:elevation="2dp"
        android:layout_height="wrap_content"
        android:focusable="true">

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

            <LinearLayout
                android:layout_margin="8dp"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical">

                <TextView
                    android:id="@+id/InstituteNameTv"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="5dp"
                    android:gravity="center_horizontal"
                    android:text="Institute Name"
                    android:textSize="20dp"
                    android:textAlignment="center"
                    android:textColor="@color/black" />

                <TextView

```

```
        android:id="@+id/InstituteNameRetrieve"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Karachi University"
        android:textStyle="bold|italic"
        android:textAlignment="center" />
<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Location"
        android:textAlignment="center"
        android:textColor="@color/black" />

<TextView
        android:id="@+id/InstituteLocationTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text=" Karachi"
        android:textAlignment="center" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Major Fields"
        android:textAlignment="center"
        android:textColor="@color/black" />

<TextView
        android:id="@+id/majorFields"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="80%"
        android:textAlignment="center" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Type"
        android:textAlignment="center"
        android:textColor="@color/black" />

<TextView
        android:id="@+id/TypeTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="College/University"
        android:textAlignment="center" />
<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
```

```

        android:text="Major Degree Program"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/MajorProgramTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="BS Education"
        android:textAlignment="center" />

    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="2dp"
        android:background="@color/colorPrimaryDark"/>

</LinearLayout>

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

Activity_view_full_book_post

```

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

<RelativeLayout
    android:id="@+id/RL"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>
<androidx.cardview.widget.CardView
    android:layout_below="@+id/myAppBarLayout"
    android:layout_width="match_parent"
    app:cardCornerRadius="5dp"
    android:layout_margin="4dp"
    android:clickable="true"
    android:id="@+id/cv_item"
    android:layout_height="wrap_content"
    android:focusable="true">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:padding="3dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/seller_profileImgIv"
            android:layout_width="75dp"
            android:layout_height="65dp"
            android:padding="8dp"
            android:layout_margin="8dp"
            android:src="@drawable/profile"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="@color/colorPrimaryDark"
            android:text="Book Posted By:"
            android:textSize="16sp"
            android:padding="8dp"
            android:textStyle="bold"
            android:layout_marginTop="15dp"
            android:layout_marginRight="5dp"/>

        <TextView
            android:id="@+id/seller_nameTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:layout_marginBottom="8dp"
            android:text="book_added_by"
            android:textColor="@color/grey"
            android:textSize="14sp"
            android:padding="8dp"
            android:textStyle="italic" />

    </LinearLayout>
    <View
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorPrimaryDark" />
    <LinearLayout
        android:layout_margin="8dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="left"
            android:layout_margin="5dp">
            <ImageView
                android:id="@+id/ImageRetrieveIv"
                android:layout_width="150dp"
                android:layout_height="250dp"
                android:src="@drawable/books"
```

```
        android:scaleType="fitXY"/>

    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:layout_margin="5dp"
        android:orientation="vertical">

        <TextView
            android:id="@+id/bookNameTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:gravity="center_horizontal"
            android:text="Book Name"
            android:textAlignment="center"
            android:textColor="@color/black" />

        <TextView
            android:id="@+id/bookNameRetrieve"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
            android:text=" Computer Science"
            android:textAlignment="center" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
            android:text="Book Author"
            android:textAlignment="center"
            android:textColor="@color/black" />

        <TextView
            android:id="@+id/bookAuthorTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:gravity="center_horizontal"
            android:text=" Nemrah"
            android:textAlignment="center" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:gravity="center_horizontal"
            android:text="Category"
            android:textAlignment="center"
            android:textColor="@color/black" />

        <TextView
            android:id="@+id/bookCatTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
```

```
        android:text="Computer Science"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Book Grade"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookClassTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text=" ICS part-II"
        android:textAlignment="center" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Price"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookPriceTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="PKR 200/-"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Description"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookDescTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Book About Computer Techniques"
        android:textAlignment="center" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Condition"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookConditionTv"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Good"
        android:textAlignment="center" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Status"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookStatusTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="PENDING"
        android:textAlignment="center" />

</LinearLayout>

</LinearLayout>

</LinearLayout>

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

Activity_view_seller

```

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/toolbar"
            android:id="@+id/main_page_toolbar">
        </include>
```

```
</com.google.android.material.appbar.AppBarLayout>
<LinearLayout
    android:id="@+id/linear1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <de.hdodenhof.circleimageview.CircleImageView
        android:id="@+id/userImg"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:src="@drawable/profile"
        android:layout_gravity="center"
        android:layout_marginTop="10dp"
        app:civ_border_color="@color/colorPrimaryDark"
        app:civ_border_width="5dp" />

</LinearLayout>
<LinearLayout
    android:id="@+id/linear2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="5dp"
    android:layout_below="@+id/linear1">

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

        <TextView
            android:id="@+id/userNameTv"
            android:layout_width="100dp"
            android:layout_height="50dp"
            android:padding="14dp"
            android:text="User Name"
            android:layout_gravity="start"
            android:textColor="@color/grey"
            android:textSize="14sp" />

        <EditText
            android:id="@+id/userNameUpdated"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="#0000"
            android:hint="abc"
            android:textAlignment="textStart"
            android:gravity="start"
            android:textStyle="italic"
            android:padding="14dp"
            android:textColor="#181737"
            android:textColorHint="#181737"
            android:textSize="14sp" />
    
```

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

    <TextView
        android:id="@+id/emailTv"
        android:layout_width="100dp"
        android:layout_height="50dp"
        android:background="#0000"
        android:gravity="start"
        android:padding="14dp"
        android:text="Email"
        android:textColor="@color/grey"
        android:textSize="14sp" />

    <TextView
        android:id="@+id/emailRetrieve"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:hint="abc@gmail.com"
        android:textAlignment="textStart"
        android:gravity="start"
        android:textStyle="italic"
        android:padding="14dp"
        android:textColor="#181737"
        android:textColorHint="#181737"
        android:textSize="14sp"/>
</LinearLayout>
/>
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#dad8d8" />

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

    <TextView
        android:id="@+id/contactTv"
        android:layout_width="100dp"
        android:layout_height="50dp"
        android:background="#0000"
        android:gravity="start"
        android:padding="14dp"
        android:text="Contact"
        android:textColor="@color/grey"
        android:textSize="14sp" />

    <EditText
        android:id="@+id/contactUpdated"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#0000"
```

```
        android:textAlignment="textStart"
        android:gravity="start"
        android:textStyle="italic"
        android:hint="0300----"
        android:inputType="number"
        android:maxLength="11"
        android:padding="14dp"
        android:textColor="#181737"
        android:textColorHint="#1b193b"
        android:textSize="14dp" />
    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#dad8d8" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/countryTv"
            android:layout_width="100dp"
            android:layout_height="match_parent"
            android:background="#0000"
            android:gravity="start"
            android:padding="14dp"
            android:text="Country"
            android:textColor="@color/grey"
            android:textSize="14sp" />
        <EditText
            android:id="@+id/countryHint"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="#0000"
            android:hint="Pakistan.."
            android:textAlignment="textStart"
            android:gravity="start"
            android:textStyle="italic"
            android:enabled="false"
            android:padding="14dp"
            android:textColor="#181737"
            android:textColorHint="#181737"
            android:textSize="14sp" />

    </LinearLayout>
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#dad8d8" />
    <LinearLayout
        android:id="@+id/layout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
```

```

<TextView
    android:id="@+id/cityTv"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:gravity="start"
    android:padding="14dp"
    android:text="City"
    android:textColor="@color/grey"
    android:textSize="14sp" />
<EditText
    android:id="@+id/cityHint"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#0000"
    android:hint="Karachi.."
    android:textAlignment="textStart"
    android:gravity="start"
    android:textStyle="italic"
    android:enabled="false"
    android:padding="14dp"
    android:textColor="#181737"
    android:textColorHint="#181737"
    android:textSize="14sp" />

</LinearLayout>
</LinearLayout>
<View
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorPrimaryDark" />

</LinearLayout>

```

Fragment_accepted_book_post

```

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

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewPosts"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>

</RelativeLayout>

```

Fragement_Public_institute

```

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

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/recycleViewPosts"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_margin="8dp"/>
```

```
</RelativeLayout>
```

Fragment_pending_post_of_books

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

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/recycleViewPosts"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_margin="8dp"/>
```

```
</RelativeLayout>
```

Fragment_Rejected_Book_post

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

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/recycleViewPosts"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_margin="8dp"/>
```

```
</RelativeLayout>
```

Fragment_private_institute

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

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/recycleViewPosts"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_margin="8dp"/>
```

```
</RelativeLayout>
```

USER SIDE CODE

USER MODEL CLASSES

BookPostModel.java

```
package com.squares.bookexchangesystem.Models;

import android.net.Uri;

public class BooksPostModel {
    String added_by;
    long added_on;
    String author;
    String bookGrade;
    String bookName;
    String category;
    String condition;
    String description;
    String edition;
    String image;
    String pid;
    String price;
    String status;
    public BooksPostModel(){}
    public BooksPostModel(
        String added_by,
        long added_on,
        String author,
        String bookGrade,
        String bookName,
        String category,
        String condition,
        String description,
        String edition,
        String image,
        String pid,
        String price,
        String status
    ) {
        this.added_by = added_by;
        this.added_on = added_on;
        this.author = author;
        this.bookGrade = bookGrade;
        this.bookName = bookName;
        this.category = category;
        this.condition = condition;
        this.description = description;
        this.edition = edition;
        this.image = image;
        this.pid = pid;
        this.price = price;
        this.status = status;
    }
    public String getImage(){
        return this.image;
    }
    public void setImage(String image){
        this.image = image;
    }
    public Uri getImageUri(){
    }
}
```

```
        return Uri.parse(this.image);
    }
    public String getBookName(){
        return this.bookName;
    }

    public void setBookName(String bookName){
        this.bookName = bookName;
    }
    public String getAdded_by(){
        return this.added_by;
    }

    public void setAdded_by(String added_by){
        this.added_by = added_by;
    }

    public long getAdded_on(){
        return this.added_on;
    }

    public void setAdded_on(long added_on){
        this.added_on = added_on;
    }
    public String getStatus(){
        return this.status;
    }

    public void setStatus(String status){
        this.status = status;
    }

    public String getAuthor(){
        return this.author;
    }

    public void setAuthor(String author){
        this.author = author;
    }

    public String getBookGrade(){
        return this.bookGrade;
    }

    public void setBookGrade(String bookGrade){
        this.bookGrade = bookGrade;
    }

    public String getCategory(){
        return this.category;
    }

    public void setCategory(String category){
        this.category = category;
    }

    public String getCondition(){
        return this.condition;
    }
```

```

    }

    public void setCondition(String condition){
        this.condition = condition;
    }

    public String getDescription(){
        return this.description;
    }

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

    public String getEdition(){
        return this.edition;
    }

    public void setEdition(String edition){
        this.edition = edition;
    }

    public String getPid(){
        return this.pid;
    }

    public void setPid(String pid){
        this.pid = pid;
    }

    public String getPrice(){
        return this.price;
    }

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

```

ChatModel.java

```

package com.squares.bookexchangesystem.Models;

public class ChatModel {
    public String type;
    public long sendAt;
    public String sendBy;
    public String text;
    public String url;
    public ChatModel() {}
    public ChatModel(String type, long sendAt, String sendBy, String text, String url)
    {
        this.type = type;
        this.sendAt = sendAt;
        this.sendBy = sendBy;
        this.url = url;
        this.text = text;
    }
}

```

ConvoModel.java

```
package com.squares.bookexchangesystem.Models;

public class ConvoModel {
    public String chatId;
    public long sendAt;
    public String sendBy;
    public String lastMessage;
    public String chattingWith;
    public ConvoModel() {}
    public ConvoModel(String chatId, long sendAt, String sendBy, String lastMessage,
String chattingWith) {
        this.chatId = chatId;
        this.sendAt = sendAt;
        this.sendBy = sendBy;
        this.lastMessage = lastMessage;
        this.chattingWith = chattingWith;
    }
}
```

EducationalInstituteModel.java

```
package com.squares.bookexchangesystem.Models;

import android.net.Uri;

public class EducationalInstituteModel {

    String Image, InstituteName, Location, Type, MajorProgram, MajorFiels;

    public EducationalInstituteModel() {}

    public EducationalInstituteModel(String image, String instituteName, String location, String type, String majorProgram, String majorFiels) {
        Image = image;
        InstituteName = instituteName;
        Location = location;
        Type = type;
        MajorProgram = majorProgram;
        MajorFiels = majorFiels;
    }

    public String getImage() {
        return Image;
    }

    public void setImage(String image) {
        Image = image;
    }

    public String getInstituteName() {
        return InstituteName;
    }

    public void setInstituteName(String instituteName) {
        InstituteName = instituteName;
    }
}
```

```

public String getLocation() {
    return Location;
}

public void setLocation(String location) {
    Location = location;
}

public String getType() {
    return Type;
}

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

public String getMajorProgram() {
    return MajorProgram;
}

public void setMajorProgram(String majorProgram) {
    MajorProgram = majorProgram;
}

public String getMajorFieis() {
    return MajorFieis;
}

public void setMajorFieis(String majorFieis) {
    MajorFieis = majorFieis;
}
}

```

ThoughtOfTheDayModel.java

```

package com.squares.bookexchangesystem.Models;

public class ThoughtOftheDay {
    String image;

    public ThoughtOftheDay() {
    }

    public ThoughtOftheDay(String image) {
        this.image = image;
    }

    public String getImage() {
        return image;
    }

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

    public String getThoughtImageUrl() {
        return image;
    }
    //    public Uri getImageUri(){
    //        return Uri.parse(this.image);
    //}
}

```

```
//      }
```

USER SIDE ADAPTERS

BookPostAdapter.java

```
package com.squares.bookexchangesystem.Adapters;

import android.app.AlertDialog;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ProgressBar;
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.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.bookexchangesystem.ChatActivity;
import com.squares.bookexchangesystem.HomeActivity;
import com.squares.bookexchangesystem.Models.BooksPostModel;
import com.squares.bookexchangesystem.R;
import com.squares.bookexchangesystem.User.User;

public class BooksPostAdapter extends FirebaseRecyclerAdapter<BooksPostModel,
BooksPostAdapter.myviewholder> {
    DatabaseReference dbUserRef =
    FirebaseDatabase.getInstance().getReference("Users");
    User sellerProfile;
    public BooksPostAdapter(@NonNull FirebaseRecyclerOptions<BooksPostModel>
options) {
        super(options);

    }
    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull BooksPostModel model)
    {
        String title = model.getBookName();

Glide.with(holder.bookImage.getContext()).load(model.getImageUri()).into(holder.bo
okImage);
        holder.title.setText(title);
        holder.bookClass.setText(model.getBookGrade());
        holder.category.setText(model.getCategory());
        holder.price.setText(model.getPrice());
```

```

        holder.buyNowBtn.setEnabled(false);
        holder.addedBy.setVisibility(View.INVISIBLE);
        holder.progressBar.setVisibility(View.VISIBLE);
        dbUserRef.child(model.getAdded_by()).addListenerForSingleValueEvent(new
ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {

        sellerProfile = snapshot.getValue(User.class);
        if(sellerProfile != null)
        {
            holder.addedBy.setText(sellerProfile.fullName);
            holder.addedBy.setVisibility(View.VISIBLE);
            if(holder.progressBar.getParent() != null) {
                ((ViewGroup)
holder.progressBar.getParent()).removeView(holder.progressBar);
            }

            holder.buyNowBtn.setEnabled(true);
        }
    }

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

    }
});;

holder.buyNowBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent(view.getContext(), ChatActivity.class);
        intent.putExtra("otherUserId", model.getAdded_by());
        view.getContext().startActivity(intent);
    }
});;

}

@NonNull
@Override
public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
{
    View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_book_posts,par
ent,false);
    return new myviewholder(view);
}

static class myviewholder extends RecyclerView.ViewHolder
{
    ImageView bookImage;
    TextView title;
    TextView category;
    TextView bookClass;
}

```

```

        TextView price;
        TextView addedBy;
        Button buyNowBtn;
        ProgressBar progressBar;

        public myviewholder(@NonNull View itemView)
        {
            super(itemView);
            bookImage = itemView.findViewById(R.id.ImageRetrieveIv);
            title = itemView.findViewById(R.id.bookNameRetrieve);
            category = itemView.findViewById(R.id.bookCatTv);
            bookClass = itemView.findViewById(R.id.bookClassTv);
            price = itemView.findViewById(R.id.bookPriceTv);
            addedBy = itemView.findViewById(R.id.seller_nameTv);
            buyNowBtn = itemView.findViewById(R.id.buyBookBtn);
            progressBar = itemView.findViewById(R.id.sellerNameprogressBar);

        }
    }
}

```

ConversationAdapter.java

```

package com.squares.bookexchangesystem.Adapters;

import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.TextView;

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

import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.bookexchangesystem.ChatActivity;
import com.squares.bookexchangesystem.Models.ConvoModel;
import com.squares.bookexchangesystem.R;
import com.squares.bookexchangesystem.User.User;

public class ConversationAdapter extends FirebaseRecyclerAdapter<ConvoModel,
ConversationAdapter.myviewholder> {

    private User convoUser;
    DatabaseReference usersDbRef =
FirebaseDatabase.getInstance().getReference("Users");
    public ConversationAdapter(@NonNull FirebaseRecyclerOptions<ConvoModel>
options) {
        super(options);
    }
}

```

```

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull ConvoModel model)
{
    usersDbRef.child(model.chattingWith).addListenerForSingleValueEvent(new
ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            convoUser = snapshot.getValue(User.class);
            if(convoUser != null) {
                holder.messageTextView.setVisibility(View.VISIBLE);
                holder.messageTextView.setText(model.lastMessage);
                holder.senderName.setText(convoUser.fullName);
            }
        }
    });

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

holder.containerLayout.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent(view.getContext(), ChatActivity.class);
        intent.putExtra("otherUserId", model.chattingWith);
        view.getContext().startActivity(intent);
    }
});
////
//switch (model.type) {
//    case "text":
//        holder.messageTextView.setVisibility(View.VISIBLE);
//        holder.messageTextView.setText(model.text);
//        holder.senderName.setText(convoUser.fullName);
//        break;
//    default:
//        break;
//}
//}

@NonNull
@Override
public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
{
    View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_display_chat_li
st,parent,false);
    return new myviewholder(view);
}

static class myviewholder extends RecyclerView.ViewHolder
{
    TextView messageTextView;
    TextView senderName;
}

```

```

        LinearLayout containerLayout;
    public myviewholder(@NonNull View itemView)
    {
        super(itemView);
        messageTextView = itemView.findViewById(R.id.message);
        senderName = itemView.findViewById(R.id.from);
        containerLayout = itemView.findViewById(R.id.linear);
    }
}

```

EducationalInstituteAdapter.java

```

package com.squares.bookexchangesystem.Adapters;

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

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

import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.squares.bookexchangesystem.Models.EducationalInstituteModel;
import com.squares.bookexchangesystem.R;

public class EducationalInstituteAdapter extends
FirebaseRecyclerAdapter<EducationalInstituteModel,
EducationalInstituteAdapter.myviewholder> {

    public EducationalInstituteAdapter(@NonNull
FirebaseRecyclerOptions<EducationalInstituteModel> options) {
        super(options);

    }
    @Override
    protected void onBindViewHolder(@NonNull
EducationalInstituteAdapter.myviewholder holder, int position, @NonNull
EducationalInstituteModel model)
    {
    //
        holder.InstituteNameTv.setText(model.getInstituteName());
        holder.LocationTv.setText(model.getLocation());
        holder.MajorFieldsTv.setText(model.getMajorFieis());
        holder.TypeTv.setText(model.getType());
        holder.MajorProgramTv.setText(model.getMajorProgram());

    }

    @NonNull
    @Override
    public EducationalInstituteAdapter.myviewholder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType)
    {

```

```

        View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_view_educationa
l_institute,parent,false);
    return new EducationalInstituteAdapter.myviewholder(view);
}

static class myviewholder extends RecyclerView.ViewHolder
{

    TextView InstituteNameTv,LocationTv, MajorFieldsTv,TypeTv,MajorProgramTv;

    public myviewholder(@NonNull View itemView)
    {
        super(itemView);
        InstituteNameTv = itemView.findViewById(R.id.InstituteNameRetrieve);
        LocationTv = itemView.findViewById(R.id.InstituteLocationTv);
        MajorFieldsTv = itemView.findViewById(R.id.majorFields);
        TypeTv = itemView.findViewById(R.id.TypeTv);
        MajorProgramTv = itemView.findViewById(R.id.MajorProgramTv);

    }
}

```

MessageAdapter.java

```

package com.squares.bookexchangesystem.Adapters;

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

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

import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.auth.FirebaseAuth;
import com.squares.bookexchangesystem.Models.ChatModel;
import com.squares.bookexchangesystem.R;
import com.squares.bookexchangesystem.User.User;

public class MessageAdapter extends FirebaseRecyclerAdapter<ChatModel,
MessageAdapter.myviewholder> {

    private User otherUser;
    private String currentUserUid =
FirebaseAuth.getInstance().getCurrentUser().getUid();
    public MessageAdapter(@NonNull FirebaseRecyclerOptions<ChatModel> options, User
otherUser) {
        super(options);
        this.otherUser = otherUser;
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull ChatModel model)

```

```

    {

        switch (model.type) {
            case "text":
                holder.messageTextView.setVisibility(View.VISIBLE);
                holder.messageTextView.setText(model.text);
                if(model.sendBy.equals(currentUserUid)) {
                    holder.senderName.setText("Me");
                } else {
                    holder.senderName.setText(otherUser.fullName);
                }

                break;
            default:
                break;
        }
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_display_chat_li
st,parent,false);
        return new myviewholder(view);
    }

    static class myviewholder extends RecyclerView.ViewHolder
    {
        TextView messageTextView;
        TextView senderName;
        public myviewholder(@NonNull View itemView)
        {
            super(itemView);
            messageTextView = itemView.findViewById(R.id.message);
            senderName = itemView.findViewById(R.id.from);
        }
    }
}

```

ThoughtImagesAdapter.java

```

package com.squares.bookexchangesystem.Adapters;

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

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

import com.bumptech.glide.Glide;
import com.firebaseio.ui.database.FirebaseRecyclerAdapter;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.squares.bookexchangesystem.Models.ThoughtOfDay;

```

```

import com.squares.bookexchangesystem.R;

import de.hdodenhof.circleimageview.CircleImageView;

public class ThoughtImagesAdapter extends
FirebaseRecyclerAdapter<ThoughtOftheDay, ThoughtImagesAdapter.myviewholder> {

    public ThoughtImagesAdapter(@NonNull FirebaseRecyclerOptions<ThoughtOftheDay>
options) {
        super(options);
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position,
@NonNull ThoughtOftheDay model)
    {

Glide.with(holder.ThoughtImagesIv.getContext()).load(model.getImage()).into(holder
.ThoughtImagesIv);
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
        View view=
LayoutInflater.from(parent.getContext()).inflate(R.layout.activity_thought_of_the_
day_images_cardview,parent,false);
        return new myviewholder(view);
    }

    static class myviewholder extends RecyclerView.ViewHolder
    {
        CardView ThoughtImagesCard;
        ImageView ThoughtImagesIv;
        public myviewholder(@NonNull View itemView)
        {
            super(itemView);
            ThoughtImagesCard=(CardView)
itemView.findViewById(R.id.thoughtCardView);
            ThoughtImagesIv=(ImageView)
itemView.findViewById(R.id.thoughtImagesRetrieve);
        }
    }
}

```

UserModel.java

```

package com.squares.bookexchangesystem.User;

public class User {
    public String fullName,
email,contact,country,city,UserProfileImage,password,added_on,updated_on;
    public boolean isSeller, isBuyer;
    public User()
    {}

    public User(String fullName, String email, String contact, String password) {
        this.fullName = fullName;
    }
}

```

```

        this.email = email;
        this.contact = contact;
        this.password=password;
    }

    public User(String fullName, String email, String contact, String country, String
city, String userProfileImage, String updated_on) {
        this.fullName = fullName;
        this.email = email;
        this.contact = contact;
        this.country = country;
        this.city = city;
        this.UserProfileImage = userProfileImage;
        this.updated_on=updated_on;
    }

    public User(String fullName, String email, String contact, String password, String
country, String city, String userProfileImage, String added_on, String updated_on,
boolean isSeller, boolean isBuyer) {
        this.fullName = fullName;
        this.email = email;
        this.contact = contact;
        this.country = country;
        this.city = city;
        UserProfileImage = userProfileImage;
        this.password = password;
        this.added_on=added_on;
        this.updated_on=updated_on;
        this.isSeller = isSeller;
        this.isBuyer = isBuyer;
    }

    public User(String fullName, String email, String contact, String password, String
country, String city, String userProfileImage, String updated_on) {
        this.fullName = fullName;
        this.email = email;
        this.contact = contact;
        this.country = country;
        this.city = city;
        UserProfileImage = userProfileImage;
        this.password = password;
        this.added_on=updated_on;
    }
}

```

USER SIDE ACTIVITY

AddBookActivity

```

package com.squares.bookexchangesystem;

import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;

```

```

import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

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

import com.google.android.gms.tasks.Continuation;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;
import com.squares.bookexchangesystem.User.User;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.HashMap;
import java.util.List;

public class AddBookActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener {

    String saveCurrentDate;
    String saveCurrentTime;
    String BookCategoryName;
    String BookName;
    String BookAuthorName;
    String BookCondition;
    String BookClassLevel;
    String BookPrice;
    String BookDescription;
    String BookEdition;
    Spinner BookCategorySpinner;
    long added_on;
//    String added_on;
    FirebaseUser current_user;
    String added_by;
    EditText
BookNameTv,BookAuthorNameTv,BookPriceTv,BookDescriptionTv,BookEditionTv;
    Spinner BookConditionSpinner,BookClassLevelSpinner;
    ImageView BookImageIv;
    Button AddBookButton;
    private Uri ImageUri;
    private static final int GalleryPick=1;
    private ProgressDialog loadingBar;
    private String productRandomKey, downloadImageUrl;
    private StorageReference BookImagesRef;
}

```

```

private DatabaseReference BookRef;
Calendar calendar;
private DatabaseReference BookCatreference;
List<String> list;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_add_book);
    calendar = Calendar.getInstance();
    init();
    adapters();
    listeners();
}

private void init() {

    // BookCategoryName = getIntent().getExtras().get("category").toString();
    // Toast.makeText(this, BookCategoryName, Toast.LENGTH_SHORT).show();
    BookImagesRef = FirebaseStorage.getInstance().getReference().child("Book
Images");
    BookRef = FirebaseDatabase.getInstance().getReference().child("Books");

    // added_on=saveCurrentDate+saveCurrentTime;
    current_user=FirebaseAuth.getInstance().getCurrentUser();
    assert current_user != null;
    added_by=current_user.getUid();
    Log.v("LOG", added_by);

    BookNameTv=(EditText) findViewById(R.id.bookNameEt);
    BookAuthorNameTv=(EditText)findViewById(R.id.authorNameEt);
    BookPriceTv=(EditText)findViewByR.id.bookPriceEt;
    BookDescriptionTv=(EditText)findViewByR.id.bookDescEt;
    BookEditionTv=(EditText)findViewByR.id.BookEditionEt;
    BookConditionSpinner=(Spinner) findViewByR.id.bookConditionSpinner;
    BookClassLevelSpinner=(Spinner) findViewByR.id.spinnerBookClass;
    loadingBar=new ProgressDialog(this);
    BookImageIv=(ImageView)findViewByR.id.uploadBookImageIv;
    BookCategorySpinner=(Spinner)findViewByR.id.spinnerBookCat;
    AddBookButton=(Button)findViewByR.id.addBookBtn;
}

private void listeners() {
    BookImageIv.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            OpenGallery();
        }
    });

    AddBookButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            ValidateBookEnterData();
        }
    });
}

```

```
}

private void ValidateBookEnterData()
{

    BookName = BookNameTv.getText().toString();
    BookAuthorName = BookAuthorNameTv.getText().toString();
    BookPrice = BookPriceTv.getText().toString();
    BookDescription= BookDescriptionTv.getText().toString();
    BookCondition= BookConditionSpinner.getSelectedItem().toString();
    BookClassLevel=BookClassLevelSpinner.getSelectedItem().toString();
    BookCategoryName=BookCategorySpinner.getSelectedItem().toString();
    BookEdition= BookEditionTv.getText().toString();

    if(BookName.isEmpty())
    {
        BookNameTv.setError("Book Name is required");
        BookNameTv.requestFocus();
    }
    else if(BookName.length()>15)
    {
        BookNameTv.setError("Book name cannot be greater than 15 characters");
        BookNameTv.requestFocus();
    }
    else if(BookAuthorName.isEmpty())
    {
        BookAuthorNameTv.setError("Author Name is required");
        BookAuthorNameTv.requestFocus();
    }
    else if(BookAuthorName.length()>15)
    {
        BookAuthorNameTv.setError("Book Author name cannot be greater than 15
characters");
        BookAuthorNameTv.requestFocus();
    }
    else if(BookPrice.isEmpty())
    {
        BookPriceTv.setError("Book Price is required");
        BookPriceTv.requestFocus();
    }
    else if(BookCondition.isEmpty())
    {
        BookConditionSpinner.requestFocus();
    }
    else if(BookCategoryName.isEmpty())
    {
        BookCategorySpinner.requestFocus();
    }
    else if(BookClassLevel.isEmpty())
    {
        BookClassLevelSpinner.requestFocus();
    }
    else if( ImageUri == null) {
        BookImageIv.requestFocus();
        Toast.makeText(AddBookActivity.this, "Error: Book cover is not
available", Toast.LENGTH_SHORT).show();
    }
}
```

```

available!", Toast.LENGTH_LONG).show();
    }
    else
    {
        // Date date = new Date();
        // added_on = date.getTime();
        StoreBookInformation();
    }
}

private void StoreBookInformation()
{
    loadingBar.setTitle("Add New Book");
    loadingBar.setMessage("Dear User, please wait while we are adding the new book.");
    loadingBar.setCanceledOnTouchOutside(false);
    loadingBar.show();

    SimpleDateFormat currentDate = new SimpleDateFormat("MMM dd, yyyy");
    saveCurrentDate = currentDate.format(calendar.getTime());

    SimpleDateFormat currentTime = new SimpleDateFormat("HH:mm:ss a");
    savecurrentTime = currentTime.format(calendar.getTime());
    added_on = new Date().getTime();
//    added_on=saveCurrentDate+" "+savecurrentTime;

    productRandomKey = BookRef.push().getKey();

    final StorageReference filePath =
BookImagesRef.child(added_by).child(ImageUri.getLastPathSegment() + "-"
+productRandomKey + ".jpg");

    final UploadTask uploadTask = filePath.putFile(ImageUri);

    uploadTask.addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e)
        {
            String message = e.toString();
            Toast.makeText/AddBookActivity.this, "Error: " + message,
Toast.LENGTH_SHORT).show();
            loadingBar.dismiss();
        }
    }).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot)
        {
            Toast.makeText/AddBookActivity.this, "Product Image uploaded
Successfully...", Toast.LENGTH_SHORT).show();

            Task<Uri> urlTask = uploadTask.continueWithTask(new
Continuation<UploadTask.TaskSnapshot, Task<Uri>>() {
                @Override
                public Task<Uri> then(@NonNull Task<UploadTask.TaskSnapshot>

```

```

task) throws Exception
{
    if (!task.isSuccessful())
    {
        throw task.getException();
    }

    downloadImageUrl = filePath.getDownloadUrl().toString();
    return filePath.getDownloadUrl();
}
).addOnCompleteListener(new OnCompleteListener<Uri>() {
    @Override
    public void onComplete(@NonNull Task<Uri> task)
    {
        if (task.isSuccessful())
        {
            downloadImageUrl = task.getResult().toString();

            Toast.makeText(AddBookActivity.this, "got the Product
image Url Successfully...", Toast.LENGTH_SHORT).show();

            SaveBookInfoToDatabase();
        }
    }
});
});

private void SaveBookInfoToDatabase()
{
    HashMap<String, Object> bookMap = new HashMap<>();

    bookMap.put("pid", productRandomKey);
    bookMap.put("added_on", added_on);
    bookMap.put("added_by", added_by);
    bookMap.put("category", BookCategoryName);
    bookMap.put("image", downloadImageUrl);
    bookMap.put("description", BookDescription);
    bookMap.put("price", BookPrice);
    bookMap.put("bookName", BookName);
    bookMap.put("edition", BookEdition);
    bookMap.put("condition", BookCondition);
    bookMap.put("author", BookAuthorName);
    bookMap.put("bookGrade", BookClassLevel);
    bookMap.put("status", "PENDING");

    BookRef.child(productRandomKey).updateChildren(bookMap)
        .addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task)
            {
                if (task.isSuccessful())
                {
                    Intent intent = new Intent(AddBookActivity.this,

```

```

ShowBookActivity.class);
        startActivity(intent);

        loadingBar.dismiss();
        Toast.makeText(AddBookActivity.this, "Book is added
successfully..", Toast.LENGTH_SHORT).show();
    }
    else
    {
        loadingBar.dismiss();
        String message = task.getException().toString();
        Toast.makeText(AddBookActivity.this, "Error: " +
message, Toast.LENGTH_SHORT).show();
    }
}
private void OpenGallery()
{
    Intent galleryIntent = new Intent();
    galleryIntent.setAction(Intent.ACTION_GET_CONTENT);
    galleryIntent.setType("image/*");
    startActivityForResult(galleryIntent, GalleryPick);
}
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode==GalleryPick && resultCode==RESULT_OK && data!=null)
    {
        ImageUri = data.getData();
        BookImageIv.setImageURI(ImageUri);
    }
}
private void adapters() {
    //bookCondition
    ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
        R.array.bookCondition, android.R.layout.simple_spinner_item);

    adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    BookConditionSpinner.setAdapter(adapter);
    BookConditionSpinner.setOnItemSelectedListener(this);

    //bookClassLevel
    ArrayAdapter<CharSequence> adapter2 = ArrayAdapter.createFromResource(this,
        R.array.bookClassLevel, android.R.layout.simple_spinner_item);

    adapter2.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    BookClassLevelSpinner.setAdapter(adapter2);
    BookClassLevelSpinner.setOnItemSelectedListener(this);

    getBookCategory();
    //bookCategorySpinner/Dropdown
//    ArrayAdapter<CharSequence> adapter3 =
ArrayAdapter.createFromResource(this,
//        R.array.bookClassLevel, android.R.layout.simple_spinner_item);
//
    adapter3.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
}

```

```

//          BookCategorySpinner.setAdapter(adapter3);
//          BookCategorySpinner.setOnItemSelectedListener(this);

}

private void getBookCategory() {
    AddBookActivity _this = this;

    DatabaseReference rootRef = FirebaseDatabase.getInstance().getReference();
    DatabaseReference eventIdRef = rootRef.child("bookCategories");
    ValueEventListener eventListener = new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            list = new ArrayList<>();
            for(DataSnapshot ds : dataSnapshot.getChildren()) {
                String bookCategory =
ds.child("category").getValue(String.class);
                list.add(bookCategory);
                Log.d("TAG", bookCategory);
            }
            Log.d("TAG", String.valueOf(list));
            list.add(0, "");
            ArrayAdapter<String> adapter3 = new ArrayAdapter(_this,
android.R.layout.simple_spinner_item, list);

adapter3.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
            BookCategorySpinner.setAdapter(adapter3);
            BookCategorySpinner.setOnItemSelectedListener(_this);
        }

        @Override
        public void onCancelled(DatabaseError databaseError) {}
    };
    eventIdRef.addListenerForSingleValueEvent(eventListener);
}

@Override
public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l)
{
    String text = adapterView.getItemAtPosition(i).toString();
    Toast.makeText(adapterView.getContext(), text, Toast.LENGTH_SHORT).show();
}

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

```

ChatActivity

```

package com.squares.bookexchangesystem;

import android.app.ProgressDialog;
import android.os.Bundle;

import com.firebaseio.ui.common.ChangeEventType;

```

```
import com.firebaseio.ui.database.ChangeEventListener;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.bookexchangesystem.Adapters.BooksPostAdapter;
import com.squares.bookexchangesystem.Adapters.MessageAdapter;
import com.squares.bookexchangesystem.Models.BooksPostModel;
import com.squares.bookexchangesystem.Models.ChatModel;
import com.squares.bookexchangesystem.Models.ConvoModel;
import com.squares.bookexchangesystem.User.User;

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

import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.widget.Toolbar;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import java.util.Date;

public class ChatActivity extends AppCompatActivity {

    private FloatingActionButton sendBtn;
    private EditText msgField;
    private Toolbar toolbar;
    private String otherUserId;
    private String chatId;

    private DatabaseReference dbRef = FirebaseDatabase.getInstance().getReference();
    private DatabaseReference convosDbRef;
    private DatabaseReference otherUserConvosDbRef;
    private DatabaseReference chatsDbRef = dbRef.child("chats");

    private String currentUserUid =
    FirebaseAuth.getInstance().getCurrentUser().getUid();

    private ProgressDialog loadingBar;

    private User otherUser;

    private MessageAdapter adapter = null;

    private RecyclerView messagesRecyclerView;

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

```

        if (savedInstanceState == null) {
            Bundle extras = getIntent().getExtras();
            if(extras == null) {
                otherUserId = null;
            } else {
                otherUserId = extras.getString("otherUserId");
            }
        } else {
            otherUserId = (String) savedInstanceState.getSerializable("sellerId");
        }
        convosDbRef =
dbRef.child("convos").child(currentUserId).child(otherUserId);
        otherUserConvosDbRef =
dbRef.child("convos").child(otherUserId).child(currentUserId);
        loadingBar = new ProgressDialog(this);
//        toolbar = findViewById(R.id.my_appBarLayout).findViewById(R.id.toolbar);
//        setSupportActionBar(toolbar);

        sendBtn = findViewById(R.id.fab_send);
        msgField = findViewById(R.id.sendMessage);
        messagesRecyclervView = findViewById(R.id.recyclerView);

        sendBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Log.v("otherUserId", otherUserId);
                Log.v("fieldText", msgField.getText().toString());
                String type = "text";
                long sendAt = new Date().getTime();
                String sendBy = currentUserId;
                String text = msgField.getText().toString().trim();
                String url = "";
                ChatModel message = new ChatModel(type, sendAt, sendBy, text, url);
                chatsDbRef.child(chatId).push().setValue(message);
                ConvoModel convoForMe = new ConvoModel(chatId, sendAt, sendBy, text,
otherUserId);
                ConvoModel convoForOther = new ConvoModel(chatId, sendAt, sendBy,
text, currentUserId);
                convosDbRef.setValue(convоМe);
                otherUserConvosDbRef.setValue(convоМe);
                msgField.setText("");
            }
        });
        init();
    }

    void init() {
        loadingBar.show();
        fethOtherUser();
    }

    private ValueEventListener getConvoListener() {
        return new ValueEventListener() {
            @Override
            public void onDataChange(DataSnapshot dataSnapshot) {

```

```

        // Get Post object and use the values to update the UI

        //AdminModel user= dataSnapshot.getValue(AdminModel.class);
        // User user = dataSnapshot.getValue(User.class);
        User user = dataSnapshot.getValue(User.class);
        ConvoModel convo = dataSnapshot.getValue(ConvoModel.class);
        if(convo != null) {
            chatId = convo.chatId;
        } else {
            chatId = chatsDbRef.push().getKey();
        }
        loadingBar.dismiss();
        setAdapter();

//                Toast.makeText(AdminLoginActivity.this, "Logged in Successfully",
Toast.LENGTH_SHORT).show();
        // ...
    }

@Override
public void onCancelled(DatabaseError databaseError) {
    // Getting Post failed, log a message
//        fbAuth.signOut();
//        Toast.makeText(AdminLoginActivity.this, "Error: You doesn't have
access", Toast.LENGTH_SHORT).show();
//        loadingBar.dismiss();
    // ...
}
};

}

void fethOtherUser() {
    dbRef.child("Users").child(otherUserId).addListenerForSingleValueEvent(new
ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        // Get Post object and use the values to update the UI

        //AdminModel user= dataSnapshot.getValue(AdminModel.class);
        // User user = dataSnapshot.getValue(User.class);
        User user = dataSnapshot.getValue(User.class);
        if(otherUser != null) {
        } else {
            otherUser = user;

convosDbRef.addListenerForSingleValueEvent(getConvoListener());

//                getSupportFragmentManager().setTitle(otherUser.fullName);
//                toolbar.setTitle(otherUser.fullName);

    }

//                Toast.makeText(AdminLoginActivity.this, "Logged in Successfully",
Toast.LENGTH_SHORT).show();

```

```

        // ...
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {
        // Getting Post failed, log a message
        //
        fbAuth.signOut();
        Toast.makeText(AdminLoginActivity.this, "Error: You doesn't have
access", Toast.LENGTH_SHORT).show();
        //
        loadingBar.dismiss();
        // ...
    });
}

void setAdapter() {
    LinearLayoutManager linearLayoutManager = new LinearLayoutManager(this);
    linearLayoutManager.setStackFromEnd(true);
    messagesRecyclcerView.setLayoutManager(linearLayoutManager);
    FirebaseRecyclerOptions<ChatModel> options =
        new FirebaseRecyclerOptions.Builder<ChatModel>()
            .setQuery(chatsDbRef.child(chatId), ChatModel.class)
            .build();
    adapter = new MessageAdapter(options, otherUser);
    messagesRecyclcerView.setAdapter(adapter);
    adapter.startListening();
    options.getSnapshots().addChangeListener(new ChangeEventListener() {
        @Override
        public void onChildChanged(@NonNull ChangeEventType type, @NonNull
DataSnapshot snapshot, int newIndex, int oldIndex) {
            Log.v("child change", "tadaaa");
            int count = messagesRecyclcerView.getAdapter().getItemCount();
            if(count > 1) {
                messagesRecyclcerView.smoothScrollToPosition(count - 1);
            }
        }
    });

    @Override
    public void onDataChanged() {
        Log.v("data cahnge", "tadaaaa");
        int count = messagesRecyclcerView.getAdapter().getItemCount();
        if(count > 1) {
            messagesRecyclcerView.smoothScrollToPosition(count - 1);
        }
    }
}

@Override
public void onError(@NonNull DatabaseError databaseError) {
    Log.v("error", "tadaaaa");
}
});

}

@Override
protected void onStart() {
    super.onStart();
}

```

```

    }

    @Override
    protected void onStop() {
        super.onStop();
        if(adapter != null) {
            adapter.stopListening();
        }
    }

}

```

PublicInstituteFragment

```

package com.squares.bookexchangesystem;

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

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.FirebaseDatabase;
import com.squares.bookexchangesystem.Adapters.EducationalInstituteAdapter;
import com.squares.bookexchangesystem.Models.EducationalInstituteModel;

/**
 * A simple {@link Fragment} subclass.
 * Use the {@link CollegeFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class CollegeFragment extends Fragment {
    RecyclerView postsRecyclcerView;
    EducationalInstituteAdapter adapter;

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

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

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

    /**
     * Use this factory method to create a new instance of
     * this fragment using the provided parameters.
     *
     * @param param1 Parameter 1.
     * @param param2 Parameter 2.
     */

```

```

    * @return A new instance of fragment CollegeFragment.
    */
    // TODO: Rename and change types and number of parameters
    public static CollegeFragment newInstance(String param1, String param2) {
        CollegeFragment fragment = new CollegeFragment();
        Bundle args = new Bundle();
        args.putString(ARG_PARAM1, param1);
        args.putString(ARG_PARAM2, param2);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View root=inflater.inflate(R.layout.fragment_college, container, false);

        postsRecyclcerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

        postsRecyclcerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

        postsRecyclcerView=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

        postsRecyclcerView.setLayoutManager(new
LinearLayoutManager(getContext()));

        FirebaseRecyclerOptions<EducationalInstituteModel> options =
                new FirebaseRecyclerOptions.Builder<EducationalInstituteModel>()

.setQuery(FirebaseDatabase.getInstance().getReference("EducationalInstitute").orde
rByChild("Type").equalTo("Public"), EducationalInstituteModel.class)
                .build();

        adapter=new EducationalInstituteAdapter(options);
        postsRecyclcerView.setAdapter(adapter);

        return root;
    }

    @Override
    public void onStart() {
        super.onStart();
        adapter.startListening();
    }
}

```

```

        @Override
        public void onStop() {
            super.onStop();
            adapter.stopListening();
        }
    }



## ConversationActivity



package com.squares.bookexchangesystem;



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.util.Log;
import android.view.View;



import com.firebaseio.ui.common.ChangeEventType;
import com.firebaseio.ui.database.ChangeEventListener;
import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.bookexchangesystem.Adapters.ConversationAdapter;
import com.squares.bookexchangesystem.Adapters.MessageAdapter;
import com.squares.bookexchangesystem.Models.ChatModel;
import com.squares.bookexchangesystem.Models.ConvoModel;



public class ConversationsActivity extends AppCompatActivity {



String currentUserUid = FirebaseAuth.getInstance().getCurrentUser().getUid();
    DatabaseReference dbRef = FirebaseDatabase.getInstance().getReference();
    DatabaseReference convosDbRef = dbRef.child("convos").child(currentUserUid);



RecyclerView listView;
    private ConversationAdapter adapter = null;



@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my_chats);
        findViewById(R.id.linear).setVisibility(View.GONE);
        listView = findViewById(R.id.recyclerView);

        setAdapter();
    }



void setAdapter() {
        LinearLayoutManager linearLayoutManager = new LinearLayoutManager(this);
        listView.setLayoutManager(linearLayoutManager);
        FirebaseRecyclerOptions<ConvoModel> options =
            new FirebaseRecyclerOptions.Builder<ConvoModel>()


```

```

        .setQuery(convosDbRef, ConvoModel.class)
        .build();
    adapter = new ConversationAdapter(options);
    listView.setAdapter(adapter);
}

@Override
protected void onStart() {
    super.onStart();
    adapter.startListening();
}

@Override
protected void onStop() {
    super.onStop();
    if(adapter != null) {
        adapter.stopListening();
    }
}
}

```

EducationalInstituteActivity

```

package com.squares.bookexchangesystem;

import android.os.Bundle;

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

import com.google.android.material.tabs.TabLayout;

public class EducationalInstituteActivity extends AppCompatActivity {
    private Toolbar mToolbar;
    private ViewPager myEIVViewPager;
    private TabLayout myEITabLayout;
    private TabAccessorAdapterForEI myEITabsAccessorAdapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_educational_institute);
        myEIVViewPager=findViewById(R.id.my_main_tabs_pager);
        myEITabsAccessorAdapter= new
        TabAccessorAdapterForEI(getSupportFragmentManager());
        myEIVViewPager.setAdapter(myEITabsAccessorAdapter);
        myEITabLayout=(TabLayout) findViewById(R.id.my_main_tabs);
        myEITabLayout.setupWithViewPager(myEIVViewPager);
    }
}

```

ForgetPasswordActivity

```

package com.squares.bookexchangesystem;

import android.app.ProgressDialog;
import android.os.Bundle;
import android.util.Patterns;

```

```

import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

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

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

public class ForgetPasswordActivity extends AppCompatActivity {

    EditText Email;
    Button Reset_password;
    private FirebaseAuth mAuth;
    private ProgressDialog loadingBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forget_password);
        mAuth= FirebaseAuth.getInstance();
        init();
        listeners();
    }

    private void listeners() {
        Reset_password.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                ResetPassword();
            }
        });
    }

    private void ResetPassword() {
        String email = Email.getText().toString().trim();
        if(email.isEmpty())
        {
            Email.setError("Email is required");
            Email.requestFocus();
            return;
        }
        if(!Patterns.EMAIL_ADDRESS.matcher(email).matches())
        {
            Email.setError("Please provide valid email");
            Email.requestFocus();
            return;
        }
        else {
            loadingBar.setTitle("Sign in");
            loadingBar.setMessage("Please wait..");
            loadingBar.setCanceledOnTouchOutside(true);
            loadingBar.show();

            mAuth.sendPasswordResetEmail(email)
                .addOnCompleteListener(new OnCompleteListener<Void>() {

```

```

        @Override
        public void onComplete(@NonNull Task<Void> task) {
            if (task.isSuccessful()) {
                Toast.makeText(ForgetPasswordActivity.this, "Check
Your Email to Reset your password", Toast.LENGTH_SHORT).show();
                loadingBar.dismiss();
            } else {
                Toast.makeText(ForgetPasswordActivity.this, "Try
Again! Something wrong happened", Toast.LENGTH_SHORT).show();
                loadingBar.dismiss();
            }
        });
    }

    private void init() {
        Email=(EditText)findViewById(R.id.reset_password_email_input);
        Reset_password=(Button)findViewById(R.id.reset_password_btn);
        loadingBar=new ProgressDialog(this);
    }
}

```

HomeActivity

```

package com.squares.bookexchangesystem;

import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.widget.Toolbar;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DataSnapshot;
import com.google.firebaseio.database.DatabaseError;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;
import com.google.firebaseio.database.ValueEventListener;
import com.squares.bookexchangesystem.User.User;

import java.text.SimpleDateFormat;
import java.util.Calendar;

public class HomeActivity extends AppCompatActivity {

    private String currentDate,currentTime;
    TextView dateDateTimeTv,currentUserTv;
    String currentUserID,CurrentUserFullName;
    private FirebaseAuth mAuth;

```

```

private FirebaseAuth user;
private DatabaseReference reference;
CardView
BookExchangeCv,DiscussionForumCv,ThoughtsOfTheDayCv,EducationalInstitutesCv;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_home);
    Toolbar toolbar = findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    init();
    // setSupportActionBar(toolbar);
    showCurrentDateTime();
    showCurrentUser();
    listeners();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.Favourites) {
        Toast.makeText(this, "Favourites", Toast.LENGTH_LONG).show();
        //startActivity(new
Intent(HomeActivity.this,FavouriteBookActivity.class));
        return true;
    }

    if (id == R.id.MyAccount) {
        startActivity(new
Intent(HomeActivity.this,UserProfileActivity.class));
        return true;
    }
    if (id == R.id.logout) {
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent(HomeActivity.this,LoginActivity.class));
        return true;
    }

    if (id == R.id.settings) {
        Toast.makeText(this, "Setting", Toast.LENGTH_LONG).show();
        return true;
    }
    return super.onOptionsItemSelected(item);
}

private void listeners() {
    BookExchangeCv.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {

```

```

        Intent intent=new Intent(HomeActivity.this,ShowBookActivity.class);
        startActivity(intent);
    });

ThoughtsOfTheDayCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeActivity.this,ShowThoughtOfTheDayActivity.class);
        startActivity(intent);
    }
});
EducationalInstitutesCv.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new
Intent(HomeActivity.this,EducationalInstituteActivity.class);
        startActivity(intent);
    }
});
}

private void showCurrentUser() {
    user=FirebaseAuth.getInstance().getCurrentUser();
    reference= FirebaseDatabase.getInstance().getReference("Users");
    currentUserID=user.getUid();
    reference.child(currentUserID).addListenerForSingleValueEvent(new
ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {

            User userProfile=snapshot.getValue(User.class);
            if(userProfile!=null)
            {
                CurrentUserFullName=userProfile.fullName;
                currentUserTv.setText("Welcome "+CurrentUserFullName+" !");
            }
        }
    });
}

@Override
public void onCancelled(@NonNull DatabaseError error) {
    Toast.makeText(HomeActivity.this,"Something
Wrong!",Toast.LENGTH_SHORT).show();
}
}

private void init() {
    dateTimeTv=(TextView)findViewById(R.id.dateTime);
    currentUserTv=(TextView)findViewById(R.id.textWelcomeUser);
    BookExchangeCv=(CardView)findViewById(R.id.bookExchange);
    DiscussionForumCv=(CardView)findViewById(R.id.discussionForum);
}

```

```

    ThoughtsOfTheDayCv=(CardView) findViewById(R.id.thoughts);

EducationalInstitutesCv=(CardView) findViewById(R.id.EducationalInstitutes);
}

private void showCurrentDateTime() {
    Calendar calendar = Calendar.getInstance();

    SimpleDateFormat _currentDate = new SimpleDateFormat("MMM dd, yyyy");
    currentDate = _currentDate.format(calendar.getTime());

    SimpleDateFormat _currentTime = new SimpleDateFormat("HH:mm:ss a");
    currentTime = _currentTime.format(calendar.getTime());

    dateTimeTv.setText(currentDate+" "+currentTime);
}

}

```

ItemClickListener

```

package com.squares.bookexchangesystem;

import android.view.View;

public interface ItemClickListner {
    void onClick(View view, int position, boolean isLongClick);
}

```

LoginActivity

```

package com.squares.bookexchangesystem;

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

import android.app.AlertDialog;
import android.content.Intent;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

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

import io.paperdb.Paper;

```

```

public class LoginActivity extends AppCompatActivity {
    private FirebaseAuth mAuth;
    private ProgressDialog loadingBar;
    EditText EmailLogin, PasswordLogin;
    Button LoginButton;

```

```

private TextView ForgetPassword;
private String parentDbName = "Users";
private CheckBox chkBoxRememberMe;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);

    mAuth= FirebaseAuth.getInstance();

    init();
    listeners();
}

private void init() {
    EmailLogin=(EditText) findViewById(R.id.Login_email_input);
    PasswordLogin=(EditText) findViewById(R.id.Login_password_input);
    loadingBar=new ProgressDialog(this);
    LoginButton=(Button) findViewById(R.id.Login_btn);
    chkBoxRememberMe = (CheckBox) findViewById(R.id.remember_me_chkb);
    ForgetPassword = (TextView) findViewById(R.id.forget_password_Link);

    Paper.init(this);
}

private void listeners() {
    LoginButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view)
        {
            AllowUserToLogin();
        }
    });
    ForgetPassword.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view)
        {
            Intent forgetPasswordIntent=new
Intent(LoginActivity.this,ForgetPasswordActivity.class);
            startActivity(forgetPasswordIntent);
        }
    });
}

private void AllowUserToLogin() {
    String email = EmailLogin.getText().toString().trim();
    String password = PasswordLogin.getText().toString().trim();
    if(email.isEmpty())
    {
        EmailLogin.setError("Email is required");
        EmailLogin.requestFocus();
        return;
    }
    if(!Patterns.EMAIL_ADDRESS.matcher(email).matches())
    {
        EmailLogin.setError("Please provide valid email");
        EmailLogin.requestFocus();
        return;
    }
}

```

```

        }

        if(password.isEmpty())
        {
            PasswordLogin.setError("Password is required");
            PasswordLogin.requestFocus();
            return;
        }
        if(password.length() != 8)
        {
            PasswordLogin.setError("Password length must be 8 characters");
            PasswordLogin.requestFocus();
            return;
        }
        else{
            loadingBar.setTitle("Login");
            loadingBar.setMessage("Please wait..");
            loadingBar.setCanceledOnTouchOutside(true);
            loadingBar.show();

            mAuth.signInWithEmailAndPassword(email,password)
                .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if(task.isSuccessful())
                        {
                            FirebaseUser
user=FirebaseAuth.getInstance().getCurrentUser();
                            assert user != null;
                            if(user.isEmailVerified())
                            {
                                SendUserToMainActivity();
                                Toast.makeText(LoginActivity.this,"Logged in
Successfully", Toast.LENGTH_SHORT).show();
                                loadingBar.dismiss();
                            }
                            else
                            {
                                user.sendEmailVerification();
                                Toast.makeText(LoginActivity.this,"Check Your
Email to verify it", Toast.LENGTH_SHORT).show();
                            }
                        }
                        else{
                            String message=task.getException().toString();
                            SendUserToRegisterActivity();

                            Toast.makeText(LoginActivity.this,"Error:"+message,Toast.LENGTH_SHORT).show();
                            loadingBar.dismiss();
                        }
                    }
                });
        }
    }

    private void SendUserToRegisterActivity() {
        Intent registerIntent=new

```

```
Intent(LoginActivity.this, RegisterActivity.class);
        startActivity(registerIntent);
    }

    private void SendUserToMainActivity() {
        Intent MainIntent=new Intent(LoginActivity.this, HomeActivity.class);
        startActivity(MainIntent);
    }
}
```

MainActivity

```
package com.squares.bookexchangesystem;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {
    Button loginButton, SignUpButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        init();
        listeners();
    }
    private void listeners() {
        loginButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                Intent in = new Intent(MainActivity.this, LoginActivity.class);
                startActivity(in);
            }
        });
        SignUpButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                Intent intent = new Intent(MainActivity.this,
                    RegisterActivity.class);
                startActivity(intent);
            }
        });
    }
    private void init() {
        loginButton=(Button)findViewById(R.id.signIn_btn);
        SignUpButton=(Button)findViewById(R.id.signUp_btn);
    }
}
```

RegisterActivity

```
package com.squares.bookexchangesystem;

import android.app.ProgressDialog;
```

```

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

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

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.bookexchangesystem.User.User;

import java.text.SimpleDateFormat;
import java.util.Calendar;

public class RegisterActivity extends AppCompatActivity {
    private Button CreateAccountButton;
    private EditText InputName, InputPhoneNumber, InputEmailNumber, InputPassword;
    private ProgressDialog loadingBar;
    private String currentDate,currentTime;
    TextView login_link;
    private FirebaseAuth mAuth;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);
        mAuth= FirebaseAuth.getInstance();
        init();
        listeners();
    }

    private void listeners() {
        CreateAccountButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                CreateAccount();
            }
        });
        login_link.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view)
            {
                Intent in = new Intent(RegisterActivity.this, LoginActivity.class);
                startActivity(in);
            }
        });
    }

    private void init() {
        CreateAccountButton = (Button) findViewById(R.id.signUp_btn);
    }
}

```

```

        InputName = (EditText) findViewById(R.id.register_name_input);
        InputPassword = (EditText) findViewById(R.id.register_password_input);
        InputPhoneNumber = (EditText) findViewById(R.id.register_contact_input);
        InputEmailNumber = (EditText) findViewById(R.id.register_email_input);
        login_link = (TextView) findViewById(R.id.login_btn_link);
        loadingBar = new ProgressDialog(this);
    }

    private void CreateAccount() {
        Calendar calendar = Calendar.getInstance();

        SimpleDateFormat _currentDate = new SimpleDateFormat("MMM dd, yyyy");
        currentDate = _currentDate.format(calendar.getTime());

        SimpleDateFormat _currentTime = new SimpleDateFormat("HH:mm:ss a");
        currentTime = _currentTime.format(calendar.getTime());

        User user=new User();
        String added_on=currentDate+" "+currentTime;
        String fullName = InputName.getText().toString().trim();
        String phone = InputPhoneNumber.getText().toString().trim();
        String email = InputEmailNumber.getText().toString().trim();
        String password = InputPassword.getText().toString().trim();
        String ImageUrl="";
        String Country="";
        String City="";
        boolean is_seller=user.isSeller;
        boolean is_buyer=user.isBuyer;
        String updated_on="";

        if(fullName.isEmpty())
        {
            InputName.setError("Full name is required");
            InputName.requestFocus();
            return;
        }
        if(fullName.length()>20)
        {
            InputName.setError("Full name cannot be greater than 20 characters");
            InputName.requestFocus();
            return;
        }
        if(phone.isEmpty())
        {
            InputPhoneNumber.setError("Phone number is required");
            InputPhoneNumber.requestFocus();
            return;
        }
        if(phone.length()!=11)
        {
            InputPhoneNumber.setError("Phone number must be of 11 digit");
            InputPhoneNumber.requestFocus();
            return;
        }
        if(email.isEmpty())
        {
            InputEmailNumber.setError("Email is required");
            InputEmailNumber.requestFocus();
            return;
        }
    }
}

```

```

        if(!Patterns.EMAIL_ADDRESS.matcher(email).matches())
        {
            InputEmailNumber.setError("Please provide valid email");
            InputEmailNumber.requestFocus();
            return;
        }
        if(password.isEmpty())
        {
            InputPassword.setError("Password is required");
            InputPassword.requestFocus();
            return;
        }
        if(password.length() != 8)
        {
            InputPassword.setError("Password length must be 8 characters");
            InputPassword.requestFocus();
            return;
        }
        else{
            loadingBar.setTitle("Creating Account");
            loadingBar.setMessage("Please wait..");
            loadingBar.setCanceledOnTouchOutside(true);
            loadingBar.show();
            mAuth.createUserWithEmailAndPassword(email,password)
                .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if(task.isSuccessful())
                        {
                            User user=new User(fullName,email,phone,password,
Country, City, ImageUrl,added_on,updated_on, is_seller, is_buyer);

                            FirebaseDatabase.getInstance().getReference("Users")
                                .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                                .setValue(user).addOnCompleteListener(new
OnCompleteListener<Void>() {
                                    @Override
                                    public void onComplete(@NonNull Task<Void> task)
{
                                        if(task.isSuccessful())
                                        {

                                            Toast.makeText(RegisterActivity.this,"Registered
Successfully",Toast.LENGTH_LONG).show();
                                            Intent registerIntent=new
Intent(RegisterActivity.this,LoginActivity.class);
                                            startActivity(registerIntent);
                                            loadingBar.dismiss();
                                        }
                                        else
                                        {
                                            Toast.makeText(RegisterActivity.this,"Register Failed",Toast.LENGTH_LONG).show();
                                            loadingBar.dismiss();
                                        }
}
                });
            }
        }
    }
}

```

```
        }
        else{
            Toast.makeText(RegisterActivity.this,"Register Failed",Toast.LENGTH_LONG).show();
            loadingBar.dismiss();
        }
    });
}
}
```

ShowBookActivity

```
package com.squares.bookexchangesystem;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.LinearLayout;

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

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.bookexchangesystem.Adapters.BooksPostAdapter;
import com.squares.bookexchangesystem.Models.BooksPostModel;

public class ShowBookActivity extends AppCompatActivity {

    LinearLayout
SellButtonlayout,HomeLayout,NotificationLayout,MyAdsLayout,MyChatLayout;
    RecyclerView postsRecyclcerView;
    BooksPostAdapter adapter;

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

        init();
        listeners();

    }

    private void listeners() {

        SellButtonlayout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new
Intent(ShowBookActivity.this,AddBookActivity.class);
                startActivity(intent);
            }
        });
    }
}
```

```

        HomeLayout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(ShowBookActivity.this,HomeActivity.class);
                startActivity(intent);
            }
        });

        MyAdsLayout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new
                Intent(ShowBookActivity.this,ShowMyBookPostActivity.class);
                startActivity(intent);
            }
        });
        MyChatLayout.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view) {
                Intent intent=new Intent(ShowBookActivity.this,
ConversationsActivity.class);
                startActivity(intent);
            }
        });
    }

    private void init() {
        SellButtonLayout=(LinearLayout) findViewById(R.id.sellBooksBtn);
        MyChatLayout=(LinearLayout) findViewById(R.id.MyChats);
        HomeLayout=(LinearLayout) findViewById(R.id.Home);
        NotificationLayout=(LinearLayout) findViewById(R.id.MyNotifications);
        MyAdsLayout=(LinearLayout) findViewById(R.id.MyBookAds);
        postsRecylicerView = findViewById(R.id.bookPostRetrieve);

        postsRecylicerView.setLayoutManager(new LinearLayoutManager(this));

        FirebaseRecyclerOptions<BooksPostModel> options =
            new FirebaseRecyclerOptions.Builder<BooksPostModel>()

.setQuery(FirebaseDatabase.getInstance().getReference("Books").orderByChild("status")
).equalTo("APPROVED"), BooksPostModel.class)
.build();

        adapter=new BooksPostAdapter(options);
        postsRecylicerView.setAdapter(adapter);
    }

    @Override
    protected void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    protected void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

ShowMyBookPostActivity

```
package com.squares.bookexchangesystem;

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

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.cardview.widget.CardView;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebaseio.database.DatabaseReference;

public class ShowMyBookPostActivity extends AppCompatActivity {

    Button RemoveBtn, ViewPostBtn;
    ImageView PostImage;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my_book_ads);
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        init();
        listeners();
    }

    private void listeners() {
        RemoveBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

            }
        });
        ViewPostBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new
                Intent(ShowMyBookPostActivity.this,ViewMyFullBookPostActivity.class);
                startActivity(intent);
            }
        });
    }

    private void init() {
        RemoveBtn=(Button)findViewById(R.id.post_remove_button);
        ViewPostBtn=(Button)findViewById(R.id.post_display_button);
        PostImage=(ImageView)findViewById(R.id.image_View_My_Ads);
    }
}
```

ShowThoughtOfTheDayActivity

```

package com.squares.bookexchangesystem;

import android.os.Bundle;
import android.widget.ImageView;

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

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.bookexchangesystem.Adapters.ThoughtImagesAdapter;
import com.squares.bookexchangesystem.Models.ThoughtOftheDay;

public class ShowThoughtOfTheDayActivity extends AppCompatActivity {

    RecyclerView ThoughtImagesRecycleView;
    ThoughtImagesAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_thoughts_of_the_day);

        init();

        ThoughtImagesRecycleView.setLayoutManager(new LinearLayoutManager(this));

        FirebaseRecyclerOptions<ThoughtOftheDay> options =
                new FirebaseRecyclerOptions.Builder<ThoughtOftheDay>()

        .setQuery(FirebaseDatabase.getInstance().getReference().child("Thought of the
Day"), ThoughtOftheDay.class)
                .build();

        adapter=new ThoughtImagesAdapter(options);
        ThoughtImagesRecycleView.setAdapter(adapter);

    }

    private void init() {
        ThoughtImagesRecycleView
        =(RecyclerView)findViewById(R.id.thoughtImagesRetrieveRecycle);
    }

    @Override
    protected void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    protected void onStop() {
        super.onStop();
        adapter.stopListening();
    }
}

```

TabAccessorForEI

```

package com.squares.bookexchangesystem;

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

import com.google.firebase.database.annotations.Nullable;

public class TabAccessorAdapterForEI extends FragmentPagerAdapter {
    public TabAccessorAdapterForEI(@NonNull FragmentManager fm) {
        super(fm);
    }

    @NonNull
    @Override
    public Fragment getItem(int position) {
        switch (position) {
            case 0:
                CollegeFragment collegeFragment = new CollegeFragment();
                return collegeFragment;
            case 1:
                UniversityFragment universityFragment = new UniversityFragment();
                return universityFragment;
            default:
                return null;
        }
    }

    @Override
    public int getCount() {
        return 2;
    }
    @Nullable
    @Override
    public CharSequence getPageTitle(int position) {

        switch (position) {
            case 0:
                return "Public";
            case 1:
                return "Private";
            default:
                return null;
        }
    }
}

```

PrivateInstituteFragment

```

package com.squares.bookexchangesystem;

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

import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

```

```

import com.firebaseio.ui.database.FirebaseRecyclerOptions;
import com.google.firebaseio.database.FirebaseDatabase;
import com.squares.bookexchangesystem.Adapters.EducationalInstituteAdapter;
import com.squares.bookexchangesystem.Models.EducationalInstituteModel;

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

    RecyclerView postsRecyclclerView;
    EducationalInstituteAdapter adapter;

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

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

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

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

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            mParam1 = getArguments().getString(ARG_PARAM1);
            mParam2 = getArguments().getString(ARG_PARAM2);
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,

```

```

        Bundle savedInstanceState) {
    // Inflate the layout for this fragment
    View root=inflater.inflate(R.layout.fragment_university, container, false);

    postsRecyclerview=(RecyclerView)
root.findViewById(R.id.recycleViewPosts);

    postsRecyclerview.setLayoutManager(new
LinearLayoutManager(getContext()));

    FirebaseRecyclerOptions<EducationalInstituteModel> options =
        new FirebaseRecyclerOptions.Builder<EducationalInstituteModel>()

.setQuery(FirebaseDatabase.getInstance().getReference("EducationalInstitute").orde
rByChild("Type").equalTo("Private"), EducationalInstituteModel.class)
        .build();

    adapter=new EducationalInstituteAdapter(options);
    postsRecyclerview.setAdapter(adapter);

}

return root;
}

@Override
public void onStart() {
    super.onStart();
    adapter.startListening();
}

@Override
public void onStop() {
    super.onStop();
    adapter.stopListening();
}
}

```

UserProfileActivity

```

package com.squares.bookexchangesystem;

import android.app.AlertDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

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

import com.google.firebase.auth.FirebaseAuth;

```

```

import com.google.firebase.auth.FirebaseAuth;
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 com.google.firebase.storage.StorageReference;
import com.squares.bookexchangesystem.User.User;

import de.hdodenhof.circleimageview.CircleImageView;

public class UserProfileActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener{
    String FullName,Email,Contact,Password,Country,City,updated_on;
    Spinner spinnerCountry;
    Spinner spinnerCity;
    String currentUserID,currentUserDate,currentTime,downloadImageUrl;
    CircleImageView UserImage;
    EditText UserName, UserContact,ErrorCity,ErrorCountry;
    TextView UserEmail;
    Button UpdateProfileButton;
    private Uri ImageUri;
    private FirebaseAuth mAuth;
    private FirebaseAuthUser user;
    private DatabaseReference reference;
    private static final int GalleryPick=1;
    private ProgressDialog loadingBar;
    private StorageReference UserProfileImagesRef;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my_profile);

        init();
        listeners();
        firebaseSignUpDataRetrieved();
        adapters();

    }

    private void listeners() {
        UserImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                OpenGallery();
            }
        });
        UpdateProfileButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                ValidateUserData();
            }
        });
    }

    private void ValidateUserData()
    {
        FullName = UserName.getText().toString();
        Email = UserEmail.getText().toString();
    }
}

```

```

Country = spinnerCountry.getSelectedItem().toString();
City= spinnerCity.getSelectedItem().toString();
Contact= UserContact.getText().toString();

if(FullName.isEmpty())
{
    UserName.setError("Full name is required");
    UserName.requestFocus();
}
else if(FullName.length()>20)
{
    UserName.setError("Full name cannot be greater than 20 characters");
    UserName.requestFocus();
}
else if(Contact.isEmpty())
{
    UserContact.setError("Phone number is required");
    UserContact.requestFocus();
    return;
}
else if(Contact.length()!=11)
{
    UserContact.setError("Phone number must be of 11 digit");
    UserContact.requestFocus();
}
else if(Country.isEmpty())
{
    Country="";
}
else if(City.isEmpty())
{
    City="";
}
// else if(ImageUri.isEmpty())
//{
//    downloadImageUrl="";
//    return;
//}
else
{
    //UpdateUserImageInformation();
}
}

private void firebaseSignUpDataRetrieved() {
    user=FirebaseAuth.getInstance().getCurrentUser();
    reference= FirebaseDatabase.getInstance().getReference("Users");
    currentUserID=user.getUid();
    reference.child(currentUserID).addValueEventListener(new
ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            User userProfile=snapshot.getValue(User.class);
            if(userProfile!=null)
            {
                FullName=userProfile.fullName;
                Email=userProfile.email;
            }
        }
    });
}

```

```

        Contact=userProfile.contact;
        Password=userProfile.password;
        Country=userProfile.country;
        City=userProfile.city;

        UserName.setText(FullName);
        UserEmail.setText(Email);
        UserContact.setText(Contact);
        ErrorCity.setText(City);
        ErrorCountry.setText(Country);

    }
}

@Override
public void onCancelled(@NonNull DatabaseError error) {
    Toast.makeText(UserProfileActivity.this,"Something Wrong!",Toast.LENGTH_SHORT).show();
}

private void adapters() {
    //country
    ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
        R.array.country, android.R.layout.simple_spinner_item);

    adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerCountry.setAdapter(adapter);
    spinnerCountry.setOnItemSelectedListener(this);

    //city
    ArrayAdapter<CharSequence> adapter2 = ArrayAdapter.createFromResource(this,
        R.array.city, android.R.layout.simple_spinner_item);

    adapter2.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerCity.setAdapter(adapter2);
    spinnerCity.setOnItemSelectedListener(this);
}

private void OpenGallery()
{
    Intent galleryIntent = new Intent();
    galleryIntent.setAction(Intent.ACTION_GET_CONTENT);
    galleryIntent.setType("image/*");
    startActivityForResult(galleryIntent, GalleryPick);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode==GalleryPick && resultCode==RESULT_OK && data!=null)
    {
        ImageUri = data.getData();
        UserImage.setImageURI(ImageUri);
    }
}

@Override

```

```

    public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l)
    {
        String text = adapterView.getItemAtPosition(i).toString();
        Toast.makeText(adapterView.getContext(), text, Toast.LENGTH_SHORT).show();
    }

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

    }

    private void init() {
        spinnerCountry = findViewById(R.id.countrySpinner);
        spinnerCity = findViewById(R.id.citySpinner);
        UserImage = (CircleImageView) findViewById(R.id.userImg);
        UserName = (EditText) findViewById(R.id.userNameUpdated);
        UserContact = (EditText) findViewById(R.id.contactUpdated);
        UserEmail=(TextView)findViewById(R.id.emailRetrieve);
        ErrorCity= (EditText) findViewById(R.id.cityHint);
        ErrorCountry= (EditText) findViewById(R.id.countryHint);
        UpdateProfileButton=(Button) findViewById(R.id.updateProfileBtn);
        loadingBar=new ProgressDialog(this);
    }
}

```

USER SIDE LAYOUT FILES CODE

Activity_books_timeline

```

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/bookPostRetrieve"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
    </androidx.recyclerview.widget.RecyclerView>

    <include layout="@layout/bottom_navigation"/>
</LinearLayout>

```

Activity_add_book

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

<LinearLayout 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"
    android:orientation="vertical">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>
    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="15dp">
            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="Add Book"
                android:textColor="@color/grey"
                android:textAlignment="center"
                android:gravity="center_horizontal"
                android:textSize="18sp"
                android:textStyle="bold"/>
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="0.8dp"
                android:background="@color/colorAccent"/>
            <TextView
                android:id="@+id/bookNametv"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="8dp"
                android:hint="Enter Book Name"
                android:background="#00000000"
                android:textColor="@color/colorPrimary"
                android:textColorHint="@color/colorPrimary"
                android:textStyle="bold|italic"
                android:textSize="12sp"/>
            <EditText
                android:id="@+id/bookNameEt"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="8dp"
                android:hint="Enter Book Name"
                android:background="#00000000"
                android:inputType="text"
                android:textColor="#000000"
                android:textColorHint="@color/grey"
```

```
        android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/authorNameTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Author Name"
    android:background="#00000000"
    android:textColor="@color/colorPrimary"
    android:textColorHint="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<EditText
    android:id="@+id/authorNameEt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Enter Author Name"
    android:background="#00000000"
    android:inputType="text"
    android:textColor="#000000"
    android:textColorHint="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/BookCatFiedTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Choose Book Category"
    android:background="#00000000"
    android:textColor="@color/colorPrimary"
    android:textColorHint="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<Spinner
    android:id="@+id/spinnerBookCat"
    android:layout_width="match_parent"
    android:layout_height="30dp"
    android:textAlignment="center"
    android:gravity="center_horizontal" />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/BookSubClassFiedTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Select Class of your Book"
    android:background="#00000000"
```

```
        android:textColor="@color/colorPrimary"
        android:textColorHint="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <Spinner
        android:id="@+id/spinnerBookClass"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:textAlignment="center"
        android:gravity="center_horizontal" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"/>
    <TextView
        android:id="@+id/editionTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:hint="Book Edition"
        android:background="#00000000"
        android:textColor="@color/colorPrimary"
        android:textColorHint="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <EditText
        android:id="@+id/BookEditionEt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:hint="Enter Edition"
        android:background="#00000000"
        android:inputType="text"
        android:textColor="#000000"
        android:textColorHint="@color/grey"
        android:textSize="12sp"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"/>
    <TextView
        android:id="@+id/bookConditionTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:hint="Book Condition"
        android:background="#00000000"
        android:textColor="@color/colorPrimary"
        android:textColorHint="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <Spinner
        android:id="@+id/bookConditionSpinner"
        android:layout_width="match_parent"
        android:layout_height="30dp"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"/>
```

```
<TextView
    android:id="@+id/bookPriceTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Book Price"
    android:background="#00000000"
    android:textColor="@color/colorPrimary"
    android:textColorHint="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<EditText
    android:id="@+id/bookPriceEt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Enter Book Price"
    android:background="#00000000"
    android:inputType="numberDecimal"
    android:textColor="#000000"
    android:textColorHint="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background= "@color/colorAccent"/>
<TextView
    android:id="@+id/bookDescTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Book Description "
    android:background="#00000000"
    android:textColor="@color/colorPrimary"
    android:textColorHint="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<EditText
    android:id="@+id/bookDescEt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:hint="Enter Book Description"
    android:background="#00000000"
    android:inputType="text"
    android:textColor="#000000"
    android:textColorHint="@color/grey"
    android:textSize="12sp"
    />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="60dp"
    android:background="@color/white">
```

```

<ImageView
    android:id="@+id/uploadBookImageIv"
    android:layout_width="150dp"
    android:layout_height="50dp"
    android:layout_gravity="right"
    android:layout_margin="20dp"
    android:background="@color/grey"
    android:padding="8dp"
    android:src="@drawable/ic_menu_gallery" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="#000000"/>
<Button
    android:id="@+id/addBookBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:text="Add Book"/>

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

```

Activity_book_post

```

<?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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="5dp"
    android:layout_margin="4dp"
    android:clickable="true"
    android:id="@+id/cv_item"
    android:layout_height="wrap_content"
    android:focusable="true">

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

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

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

```

```
        android:layout_width="100dp"
        android:layout_height="40dp"
        android:src="@drawable/profile"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="@color/colorPrimaryDark"
        android:text="Book Posted By:"
        android:textSize="16sp"
        android:textStyle="bold"
        android:layout_margin="8dp" />

    <ProgressBar
        android:id="@+id/sellerNameprogressBar"
        style="?android:attr/progressBarStyle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1" />

    <TextView
        android:id="@+id/seller_nameTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:text="Seller Name"
        android:textColor="@color/grey"
        android:textSize="14sp"
        android:textStyle="italic" />

</LinearLayout>

<LinearLayout
    android:layout_margin="8dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="200dp"
        android:layout_gravity="left"
        android:layout_margin="5dp">
        <ImageView
            android:id="@+id/ImageRetrieveIv"
            android:layout_width="150dp"
            android:layout_height="200dp"
            android:src="@drawable/cs_book"
            android:scaleType="fitXY"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="200dp"
        android:layout_gravity="right"
        android:layout_margin="5dp"
        android:orientation="vertical">
        <TextView
```

```
        android:id="@+id/bookNameTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Book Name"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookNameRetrieve"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text=" Computer Science"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Category"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookCatTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Computer Science"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Class Level"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookClassTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text=" ICS part-II"
        android:textAlignment="center" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Price"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/bookPriceTv"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="PKR 200/-"
        android:textAlignment="center" />

    <TextView
        android:id="@+id/moreInfoTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="More Details"
        android:textAlignment="center"
        android:textColor="#0000FF" />

    </LinearLayout>

    <Button
        android:id="@+id/buyBookBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:text="Buy"/>
</LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

```

Activity_display_chatlist

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/linear"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_margin="10dp">
    <LinearLayout
        android:paddingBottom="10dp"
        android:gravity="center"
        android:id="@+id/linearDate"
        android:layout_gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <androidx.cardview.widget.CardView
            android:id="@+id/cardDate"

```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:cardCornerRadius="10dp">
    <TextView
        android:id="@+id/textDate"
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:layout_marginTop="5dp"
        android:layout_marginBottom="5dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="11/06/2020">

    </TextView>

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

<TextView
    android:id="@+id/from"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="53dp"
    android:textStyle="italic"
    android:text="Name"
    android:layout_marginLeft="53dp" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/linear2"
    android:layout_gravity="center_vertical"
    android:orientation="horizontal">
    <de.hdodenhof.circleimageview.CircleImageView
        android:layout_width="40dp"
        android:id="@+id/imageView"
        android:layout_height="40dp"
        android:layout_gravity="top"
        android:src="@drawable/profile"/>
    <androidx.cardview.widget.CardView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/cardView"
        android:paddingTop="5dp"
        android:paddingBottom="5dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"
        app:cardCornerRadius="15dp">

        <TextView
            android:id="@+id/message"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="5dp"
            android:layout_marginTop="10dp"
            android:layout_marginEnd="5dp"
            android:layout_marginBottom="10dp"
            android:text="message"
            android:textColor="@color/black"
```

```

        android:visibility="gone" />

    <ImageView
        android:id="@+id/imageMessage"
        android:layout_width="230dp"
        android:layout_height="wrap_content"
        android:scaleType="center"
        android:src="@drawable/ic_menu_gallery"
        android:visibility="gone"/>
    </androidx.cardview.widget.CardView>

</LinearLayout>

<TextView
    android:id="@+id/time"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="53dp"
    android:textSize="10dp"
    android:textStyle="italic"
    android:text="12:00AM"
    android:layout_marginLeft="53dp" />

</LinearLayout>
```

Activity_educational_institute

```

<?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">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        tools:ignore="MissingConstraints">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
        <com.google.android.material.tabs.TabLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/my_main_tabs"
            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">
        </com.google.android.material.tabs.TabLayout>
    </com.google.android.material.appbar.AppBarLayout>
    <androidx.viewpager.widget.ViewPager
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/my_main_tabs_pager"
        android:layout_below="@+id/myAppBarLayout"
        tools:ignore="MissingConstraints">
        </androidx.viewpager.widget.ViewPager>
    </androidx.constraintlayout.widget.ConstraintLayout>
```

Activity_forget_password

```
<?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"
    android:background="@drawable/home_bg">
    <ImageView
        android:id="@+id/app_logo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="90dp"
        android:src="@drawable/applogo"
        />
    <TextView
        android:id="@+id/loginPage"
        android:layout_marginTop="5dp"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:fontFamily="serif"
        android:layout_below="@id/app_logo"
        android:gravity="center"
        android:text="Book Exchange System"
        android:textAlignment="center"
        android:textColor="@color/white"
        android:textColorHighlight="@color/colorAccent"
        android:textSize="30sp"
        android:textStyle="bold" />
    <View
        android:layout_width="350dp"
        android:layout_height="1.5dp"
        android:background="@color/colorPrimaryDark"
        android:layout_below="@id/loginPage"
        android:layout_marginTop="5dp"
        android:layout_marginBottom="5dp"
        android:layout_centerInParent="true"/>
    <EditText
        android:id="@+id/reset_password_email_input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/loginPage"
        android:layout_marginLeft="15dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="15dp"
        android:background="@drawable/input_design_login"
        android:drawableStart="@android:drawable/ic_dialog_email"
        android:fontFamily="serif"
        android:hint=" Email"
        android:inputType="textEmailAddress"
        android:padding="10dp"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        android:textSize="16sp"
        android:drawableLeft="@android:drawable/ic_dialog_email" />
```

```
<Button
    android:id="@+id/reset_password_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/reset_password_email_input"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/buttons"
    android:padding="10dp"
    android:text="Reset password"
    android:textStyle="bold"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="15sp"
    app:backgroundTint="@color/colorPrimaryDark" />
```

```
</RelativeLayout>
```

Activity_home

```
<?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="@drawable/home_bg">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>
    <RelativeLayout
        android:id="@+id/rL1"
        android:layout_weight="1"
        android:padding="16dp"
        android:layout_width="match_parent"
        android:layout_height="200dp">

        <TextView
            android:id="@+id/textWelcomeUser"
            android:text="Welcomes user"
            android:textSize="20sp"
            android:textStyle="bold"
            android:layout_marginRight="10dp"
            android:layout_above="@+id/dateTime"
            android:textColor="@android:color/white"
            android:layout_width="wrap_content"
```

```
        android:layout_alignParentRight="true"
        android:layout_height="wrap_content"
        android:layout_marginEnd="10dp"
        android:layout_alignParentEnd="true" />
<TextView
    android:id="@+id/dateTime"
    android:text="Tuesday, 6-ct-2020"
    android:textSize="13sp"
    android:textColor="@android:color/white"
    android:layout_marginRight="10dp"
    android:textStyle="bold|italic"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_marginBottom="15dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="10dp"
    android:layout_alignParentEnd="true" />

</RelativeLayout>

<GridLayout
    android:id="@+id/mainGrid"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="6"
    android:layout_gravity="bottom"
    android:alignmentMode="alignMargins"
    android:columnCount="2"
    android:columnOrderPreserved="false"
    android:padding="5dp"
    android:rowCount="2">

    <!-- Row 1 -->
    <!-- Column 1 -->
    <!-- Column 2 -->
    <androidx.cardview.widget.CardView
        android:id="@+id/bookExchange"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_row="0"
        android:layout_rowWeight="1"
        android:layout_column="0"
        android:layout_columnWeight="1"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginBottom="15dp"
        app:cardCornerRadius="8dp"
        app:cardElevation="8dp">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal|center_vertical"
            android:layout_margin="16dp"
            android:orientation="vertical">
```

```
<ImageView  
    android:id="@+id/bookImg"  
    android:layout_width="wrap_content"  
    android:layout_height="49dp"  
    android:layout_gravity="center_horizontal"  
    android:src="@drawable/books" />  
  
<TextView  
    android:id="@+id/bookID"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center_horizontal"  
    android:text="Book Exchange"  
    android:textAlignment="center"  
    android:textColor="@android:color/black"  
    android:textSize="15sp"  
    android:textStyle="bold" />  
  
</LinearLayout>  
  
</androidx.cardview.widget.CardView>  
  
<androidx.cardview.widget.CardView  
    android:id="@+id/discussionForum"  
    android:layout_width="0dp"  
    android:layout_height="0dp"  
    android:layout_rowWeight="1"  
    android:layout_columnWeight="1"  
    android:layout_marginLeft="5dp"  
    android:layout_marginRight="8dp"  
    android:layout_marginBottom="16dp"  
    app:cardCornerRadius="8dp"  
    app:cardElevation="8dp">  
  
<LinearLayout  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="center_horizontal|center_vertical"  
    android:layout_margin="16dp"  
    android:orientation="vertical">  
  
<ImageView  
    android:layout_width="wrap_content"  
    android:layout_height="49dp"  
    android:layout_gravity="center_horizontal"  
    android:src="@drawable/discussionforum" />  
  
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center_horizontal"  
    android:text="Discussion Forum"  
    android:textAlignment="center"  
    android:textColor="@android:color/black"  
    android:textSize="15sp"  
    android:textStyle="bold" />  
  
</LinearLayout>
```

```
</androidx.cardview.widget.CardView>
<!-- Row 2 -->

<!-- Column 1 -->
<androidx.cardview.widget.CardView
    android:id="@+id/thoughts"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginLeft="5dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="49dp"
            android:layout_gravity="center_horizontal"
            android:src="@drawable/thoughts" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:gravity="center_horizontal"
            android:text="Thought of the Day"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>

<!-- Column 2 -->
<androidx.cardview.widget.CardView
    android:id="@+id/EducationalInstitutes"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_marginLeft="5dp"
    android:layout_marginRight="8dp"
    android:layout_marginBottom="15dp"
    app:cardCornerRadius="8dp"
    app:cardElevation="8dp"
    android:layout_marginStart="5dp"
    android:layout_marginEnd="8dp">

    <LinearLayout
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="49dp"
            android:layout_gravity="center_horizontal"
            android:src="@drawable/educational_institutes" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:gravity="center_horizontal"
            android:text="Educational Institutes"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>

</androidx.cardview.widget.CardView>

<!-- Row 2 -->

</GridLayout>

```

</LinearLayout>

Activity_login

```

<?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"
    android:background="@drawable/login"
    tools:context=".LoginActivity">

    <ImageView
        android:id="@+id/login_applogo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:src="@drawable/applogo"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="120dp"
        />
    <TextView
        android:id="@+id/wel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="WELCOME TO BOOK EXCHANGE APPLICATION"
        android:layout_centerInParent="true"
        android:textColor="@color/white"
        />

```

```
        android:layout_below="@+id/login_applogo"
        android:textSize="20dp"
        android:textAlignment="center"
        android:layout_margin="20dp"
        android:gravity="center_horizontal"
        android:textStyle="bold|italic"/>
<EditText
    android:id="@+id/login_email_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/wel"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
/>
<EditText
    android:id="@+id/login_password_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/login_email_input"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="6dp"
    android:hint="Password"
    android:inputType="textPassword"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
/>
<LinearLayout
    android:id="@+id/linear_layout_1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_below="@+id/login_password_input"
    android:layout_marginTop="5dp"
    >
<CheckBox
    android:id="@+id/remember_me_chkb"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:visibility="invisible"
    android:text="Remember me"
    android:textColor="@color/white"

    android:gravity="center_vertical"
    android:textSize="14.5sp"
```

```
        android:layout_marginLeft="17dp"
    />

    <TextView
        android:id="@+id/forget_password_link"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Forget Password?"
        android:textColor="@color/white"
        android:textSize="14.5sp"
        android:layout_marginLeft="70dp"
    />

</LinearLayout>

<Button
    android:id="@+id/login_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/linear_layout_1"
    android:layout_marginLeft="15dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/buttons"
    android:padding="16dp"
    android:text="Login"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="15sp"
    app:backgroundTint="@color/colorPrimary" />
```

```
</RelativeLayout>
```

Activity_main

```
<?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"
    android:background="@drawable/welcome"
    tools:context=".MainActivity" >

    <ImageView
        android:id="@+id/app_logo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="60dp"
        android:src="@drawable/applogo"
    />
```

```
<TextView
    android:id="@+id/app_slogan"
    android:layout_width="248dp"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_marginEnd="80dp"
    android:layout_marginTop="142dp"
    android:text="Find, Discover, Choose and Buy old books online."
    android:textAlignment="center"
    android:textColor="@color/white"
    android:textSize="18sp"
    android:textStyle="italic"
    android:layout_centerInParent="true"
    android:layout_marginRight="80dp"
    android:gravity="center_horizontal" />

<Button
    android:id="@+id/signIn_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentBottom="true"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginBottom="10dp"
    android:padding="20dp"
    android:text="Already have an Account? Login"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="14sp"
    android:background="@drawable/buttons"
    android:textAlignment="center"
    app:backgroundTint="@color/colorPrimary"
    android:gravity="center_horizontal" />

<Button
    android:id="@+id/signUp_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/signIn_btn"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginBottom="5dp"
    android:background="@drawable/input_design"
    android:padding="20dp"
    android:textAlignment="center"
    android:text="Join Now"
    android:textAllCaps="false"
    android:textColor="@android:color/white"
    android:textSize="14sp"
    app:backgroundTint="@color/transparent"
    android:gravity="center_horizontal" />

</RelativeLayout>
```

Activity_main_discussion_forum

```
<?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">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_below="@+id/myAppBarLayout"
        android:layout_above="@+id/linear"
        android:layout_height="match_parent"
        android:id="@+id/recyclerView"/>

    <LinearLayout
        android:id="@+id/linear"
        android:gravity="center_vertical"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_alignParentBottom="true"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/add_data"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="5dp"
            android:layout_marginLeft="5dp"
            android:src="@android:drawable/ic_menu_add" />
        <EditText
            android:id="@+id/sendMessage"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:background="@drawable/bg_edit_text"
            android:padding="10dp"
            android:layout_marginStart="5dp"
            android:layout_weight="1"
            android:autofillHints=""
            android:layout_marginLeft="5dp" />

        <com.google.android.material.floatingactionbutton.FloatingActionButton
            android:id="@+id/fab_send"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="5dp"
            android:layout_marginEnd="5dp"
            android:src="@android:drawable/ic_menu_send" />
    
```

```

        android:tint="#FF81FD03"
        app:fabCustomSize="47dp" />

    </LinearLayout>

</RelativeLayout>

```

Activity_my_book_ads

```

<?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"
    android:layout_marginStart="15dp"
    android:layout_marginBottom="20dp"
    android:layout_marginTop="20dp"
    android:layout_marginEnd="15dp">
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <TextView
            android:id="@+id/title_My_Ads"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="My Ads"
            android:textColor="@color/black"
            android:textSize="20dp"
            android:textStyle="italic|bold"
            android:textAlignment="center"
            android:gravity="center_horizontal">
        </TextView>
        <ImageView
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:id="@+id/image_View_My_Ads"
            android:layout_below="@+id/title_My_Ads"
            android:scaleType="centerCrop">
        </ImageView>
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:layout_below="@+id/image_View_My_Ads"
            android:orientation="horizontal">
            <Button
                android:layout_width="wrap_content"
                android:layout_height="40dp"
                android:id="@+id/post_display_button"
                android:text="Display"
                android:textSize="15sp"
                android:textAllCaps="false"
                android:background="@color/teal_700"
                android:layout_marginStart="8dp"
                android:textColor="@color/white"
                android:layout_marginTop="12dp"

```

```
        android:textStyle="italic"
        android:layout_marginLeft="5dp">

    </Button>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="40dp"
        android:id="@+id/post_remove_button"
        android:text="Remove"
        android:textSize="15sp"
        android:textAllCaps="false"
        android:background="@color/teal_700"
        android:layout_marginStart="8dp"
        android:textColor="@color/white"
        android:layout_marginTop="12dp"
        android:textStyle="italic"
        android:layout_marginLeft="5dp">

    </Button>

</LinearLayout>
</RelativeLayout>
```

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

Activity_my_chat

```
<?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">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/ThemeOverlay.AppCompat">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_below="@+id/myAppBarLayout"
        android:layout_above="@+id/linear"
        android:layout_height="match_parent"
        android:id="@+id/recyclerView"/>

    <LinearLayout
        android:id="@+id/linear"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_alignParentBottom="true"
        android:gravity="center_vertical"
        android:orientation="horizontal">
```

```

<EditText
    android:id="@+id/sendMessage"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:autofillHints=""
    android:background="@drawable/bg_edit_text"
    android:padding="10dp" />

<ImageView
    android:id="@+id/add_data"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@android:drawable/ic_menu_add" />

<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:id="@+id/fab_send"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="5dp"
    android:layout_marginEnd="5dp"
    android:src="@android:drawable/ic_menu_send"
    android:tint="#FF81FD03"
    app:fabCustomSize="47dp" />

</LinearLayout>

</RelativeLayout>

```

Activity_my_profile

```

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>
    <LinearLayout
        android:id="@+id/linear1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/userImg"
            android:layout_width="150dp"
            android:layout_height="150dp"
            android:src="@drawable/profile"

```

```
        android:layout_gravity="center"
        android:layout_marginTop="10dp"
        app:civ_border_color="@color/colorPrimaryDark"
        app:civ_border_width="5dp" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Change Picture"
        android:textAlignment="center"
        android:gravity="center_horizontal" />
</LinearLayout>
<LinearLayout
    android:id="@+id/linear2"
    android:layout_width="match_parent"
    android:layout_height="239dp"
    android:orientation="vertical"
    android:layout_below="@+id/linear1">

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

        <TextView
            android:id="@+id/userNameTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:padding="14dp"
            android:text="User Name"
            android:layout_gravity="start"
            android:textColor="@color/grey"
            android:textSize="14sp" />

        <EditText
            android:id="@+id/userNameUpdated"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="#0000"
            android:gravity="end"
            android:hint="abc"
            android:padding="14dp"
            android:textColor="#181737"
            android:textColorHint="#181737"
            android:textSize="14sp" />
    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#dad8d8" />

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

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

```
        android:layout_height="wrap_content"
        android:background="#0000"
        android:gravity="start"
        android:padding="14dp"
        android:text="Email"
        android:textColor="@color/grey"
        android:textSize="14sp" />

    <TextView
        android:id="@+id/emailRetrieve"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:gravity="right"
        android:hint="abc@gmail.com"
        android:textColor="#181737"
        android:textColorHint="#1b193b"
        android:textSize="14sp" />
</LinearLayout>
/>
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#dad8d8" />

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

    <TextView
        android:id="@+id/contactTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:gravity="start"
        android:padding="14dp"
        android:text="Contact"
        android:textColor="@color/grey"
        android:textSize="14sp" />

    <EditText
        android:id="@+id/contactUpdated"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:gravity="end"
        android:hint="0300----"
        android:inputType="number"
        android:maxLength="11"
        android:padding="14dp"
        android:textColor="#181737"
        android:textColorHint="#1b193b"
        android:textSize="14dp" />
</LinearLayout>

<View
    android:layout_width="match_parent"
```

```
        android:layout_height="1dp"
        android:background="#dad8d8" />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/countryTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:gravity="start"
        android:padding="14dp"
        android:text="Country"
        android:textColor="@color/grey"
        android:textSize="14sp" />
    <EditText
        android:id="@+id/countryHint"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:enabled="false"
        android:inputType="none"
        android:textColor="@color/brick_red" />
    <Spinner
        android:id="@+id/countrySpinner"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="end"
        android:padding="14dp"
        android:textColor="#181737"
        android:textSize="14dp" />

</LinearLayout>
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="#dad8d8" />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/cityTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="start"
        android:padding="14dp"
        android:text="City"
        android:textColor="@color/grey"
        android:textSize="14sp" />
    <EditText
        android:id="@+id/cityHint"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0000"
        android:enabled="false"
```

```

        android:inputType="none"
        android:textColor="@color/brick_red" />
    <Spinner
        android:id="@+id/citySpinner"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="end"
        android:padding="14dp"
        android:textColor="#181737"
        android:textColorHint="#1b193b"
        android:textSize="14sp" />

    </LinearLayout>
    <View
        android:layout_width="match_parent"
        android:layout_height="3dp"
        android:background="@color/grey" />
</LinearLayout>

<Button
    android:id="@+id/updateProfileBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginEnd="10dp"
    android:layout_marginRight="10dp"
    android:text="Update" />

```

</LinearLayout>

Activity_register

```

<?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"
    android:background="@drawable/loginbg"
    tools:context=".RegisterActivity">

    <ImageView
        android:id="@+id/register_applogo"
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:src="@drawable/applogo"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp"
        />
    <TextView
        android:id="@+id/wel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="WELCOME TO BOOK EXCHANGE APPLICATION"
        android:layout_centerInParent="true"
        android:textColor="@color/white"
        android:layout_below="@+id/register_applogo"
        android:textSize="20dp"
        android:textAlignment="center"
        />

```

```
        android:layout_margin="10dp"
        android:gravity="center_horizontal"
        android:textStyle="bold|italic"/>
<EditText
    android:id="@+id/register_name_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/wel"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:hint="Name"
    android:inputType="text"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
/>
<EditText
    android:id="@+id/register_contact_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/register_name_input"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:hint="Phone Number"
    android:inputType="number"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
    android:layout_marginTop="6dp"
/>
<EditText
    android:id="@+id/register_email_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/register_contact_input"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:layout_marginTop="6dp"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:textColor="@color/white"
    android:textColorHint="@color/white"
    android:textSize="16sp"
/>
<EditText
    android:id="@+id/register_password_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/register_email_input"
    android:background="@drawable/input_design_login"
    android:padding="16dp"
```

```

        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:layout_marginTop="6dp"
        android:hint="Password"
        android:inputType="textPassword"
        android:textColor="@color/white"
        android:textColorHint="@color/white"
        android:textSize="16sp"
    />

<Button
    android:id="@+id/signUp_btn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/register_password_input"
    android:layout_marginTop="5dp"
    android:layout_marginLeft="15dp"
    android:layout_marginRight="15dp"
    android:background="@drawable/buttons"
    android:padding="16dp"
    android:textAllCaps="false"
    android:textSize="15sp"
    android:text="Create Account"
    android:textColor="@android:color/white"
    app:backgroundTint="@color/colorPrimary"/>

<TextView
    android:id="@+id/login_btn_link"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/signUp_btn"
    android:layout_margin="1dp"
    android:layout_centerInParent="true"
    android:fontFamily="serif"
    android:text="Already have an account? Login"
    android:textColor="@color/white"
    android:textSize="16sp"
    android:textStyle="bold|italic" />

</RelativeLayout>

```

Activity_register

```

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

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="3dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/input_design"
        android:layout_margin="5dp"
        android:orientation="horizontal">
            <EditText

```

```

        android:id="@+id/search_barEt"
        android:layout_width="230dp"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginLeft="10dp"
        android:padding="8dp"
        android:background="#00000000"
        android:hint="Search books"
        android:inputType="text"
        android:letterSpacing="0.1"
        android:textColor="#000000"
        android:textColorHint="@color/grey"
        android:textSize="13dp"
        android:layout_marginStart="10dp"
        android:layout_marginRight="10dp"/>
    <ImageView
        android:id="@+id/searchIconIv"
        android:layout_width="40dp"
        android:layout_height="50dp"
        android:src="@drawable/search"/>
    <Spinner
        android:id="@+id/search_filter_spinner"
        android:layout_width="90dp"
        android:layout_height="50dp" />
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.6dp"
    android:orientation="horizontal"
    android:background="@color/colorPrimaryDark"/>
</LinearLayout>

```

Activity_search_bar

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="3dp">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/input_design"
        android:layout_margin="5dp"
        android:orientation="horizontal">
        <EditText
            android:id="@+id/search_barEt"
            android:layout_width="230dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginLeft="10dp"
            android:padding="8dp"
            android:background="#00000000"
            android:hint="Search books"

```

```

        android:inputType="text"
        android:letterSpacing="0.1"
        android:textColor="#000000"
        android:textColorHint="@color/grey"
        android:textSize="13dp"
        android:layout_marginStart="10dp"
        android:layout_marginRight="10dp"/>
    <ImageView
        android:id="@+id/searchIconIv"
        android:layout_width="40dp"
        android:layout_height="50dp"
        android:src="@drawable/search"/>
    <Spinner
        android:id="@+id/search_filter_spinner"
        android:layout_width="90dp"
        android:layout_height="50dp" />
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.6dp"
    android:orientation="horizontal"
    android:background="@color/colorPrimaryDark"/>
</LinearLayout>
```

Activity_thought_images_cardview

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/thoughtCardView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="5dp"
    android:padding="16dp">

    <ImageView
        android:id="@+id/thoughtImagesRetrieve"
        android:layout_width="match_parent"
        android:layout_height="250dp"
        android:background="@drawable/input_design"
        android:layout_margin="10dp"
        android:scaleType="fitXY"
        android:padding="10dp" />
</androidx.cardview.widget.CardView>
```

Activity_thought_of_the_day

```

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

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
```

```
        layout="@layout/tool_bar_main"
        android:id="@+id/main_page_toolbar">
    </include>
</com.google.android.material.appbar.AppBarLayout>

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/thoughtImagesRetrieveRecycle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp">

        </androidx.recyclerview.widget.RecyclerView>

    </ScrollView>
```

```
</LinearLayout>
```

Activity_view_educational_institute

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

<RelativeLayout
    android:id="@+id/RL"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <androidx.cardview.widget.CardView
        android:layout_below="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        app:cardCornerRadius="10dp"
        android:layout_margin="4dp"
        android:clickable="true"
        android:id="@+id/cv_item"
        android:elevation="2dp"
        android:layout_height="wrap_content"
        android:focusable="true">

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

            <LinearLayout
                android:layout_margin="8dp"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="vertical">

                <TextView
```

```
        android:id="@+id/InstituteNameTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Institute Name"
        android:textSize="20dp"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/InstituteNameRetrieve"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Karachi University"
        android:textStyle="bold|italic"
        android:textAlignment="center" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Location"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/InstituteLocationTv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text=" Karachi"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:gravity="center_horizontal"
        android:text="Major Fields"
        android:textAlignment="center"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/majorFields"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="80%"
        android:textAlignment="center" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="Type"
        android:textAlignment="center"
        android:textColor="@color/black" />
```

```

        <TextView
            android:id="@+id/TypeTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
            android:text="College/University"
            android:textAlignment="center" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
            android:text="Major Degree Program"
            android:textAlignment="center"
            android:textColor="@color/black" />

        <TextView
            android:id="@+id/MajorProgramTv"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center_horizontal"
            android:text="BS Education"
            android:textAlignment="center" />

    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="2dp"
        android:background="@color/colorPrimaryDark"/>

</LinearLayout>

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

```

Activity_view_my_book_ad

```

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

<LinearLayout 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"
    android:orientation="vertical">
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/myAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <include
            layout="@layout/tool_bar_main"
            android:id="@+id/main_page_toolbar">
        </include>
    </com.google.android.material.appbar.AppBarLayout>

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

```

```
<ImageView
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:background="@drawable/input_design"
    android:id="@+id/image_View_retrieve_my_image_ad"
    android:scaleType="fitXY">
</ImageView>
</RelativeLayout>
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="15dp">
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="My Ad Details"
    android:textColor="@color/grey"
    android:textAlignment="center"
    android:gravity="center_horizontal"
    android:textSize="18sp"
    android:textStyle="bold"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.8dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/bookNameRetrieveTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:text="Book Name"
    android:textColor="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<TextView
    android:id="@+id/bookNameRetrieveEt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:textColor="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/authorNameRetrieveTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:text="Author Name"
    android:background="#0000"
    android:textColor="@color/colorPrimary"
```

```
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
<TextView
    android:id="@+id/authorNameRetrieveEt"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:textColor="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/BookSubClassFieldRetrieveTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:text="Book Class"
    android:textColor="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<TextView
    android:id="@+id/spinnerBookClassRetrieve"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:textColor="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
    android:id="@+id/editionRetrieveTv"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:text="Book Edition"
    android:textColor="@color/colorPrimary"
    android:textStyle="bold|italic"
    android:textSize="12sp"/>
<TextView
    android:id="@+id/editionSpinnerRetrieve"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:background="#0000"
    android:textColor="@color/grey"
    android:textSize="12sp"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>
<TextView
```

```
        android:id="@+id/bookConditionRetrieveTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:background="#0000"
        android:text="Book Condition"
        android:textColor="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <TextView
        android:id="@+id/bookConditionSpinnerRetrieve"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:background="#0000"
        android:textColor="@color/grey"
        android:textSize="12sp"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background="@color/colorAccent"/>
    <TextView
        android:id="@+id/bookPriceRetrieveTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:background="#0000"
        android:text="Book Price"
        android:textColor="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <TextView
        android:id="@+id/bookPriceRetrieveEt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:background="#0000"
        android:textColor="@color/grey"
        android:textSize="12sp"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0.5dp"
        android:background= "@color/colorAccent"/>
    <TextView
        android:id="@+id/bookDescRetrieveTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        android:background="#0000"
        android:text="Book Description"
        android:textColor="@color/colorPrimary"
        android:textStyle="bold|italic"
        android:textSize="12sp"/>
    <TextView
        android:id="@+id/bookDescRetrieveEt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```

        android:layout_margin="8dp"
        android:background="#0000"
        android:textColor="@color/grey"
        android:textSize="12sp"
    />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="@color/colorAccent"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0.5dp"
    android:background="#000000"/>
<Button
    android:id="@+id/edit_Post_details_of_ads"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:layout_gravity="center"
    android:text="Edit Post"/>

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

Activity_bottom_navigation

```

<?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">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:orientation="horizontal"
        android:layout_margin="5dp">

        <LinearLayout
            android:id="@+id/Home"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_gravity="center"
            android:layout_margin="5dp">

            <ImageView
                android:layout_width="50dp"
                android:layout_height="35dp"
                android:src="@drawable/home"
                android:layout_gravity="center"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Home"
    android:textAlignment="center"
    android:layout_gravity="center"
    android:textColor="@android:color/black"
    android:textSize="15sp"
    android:textStyle="bold"/>

</LinearLayout>

<LinearLayout
    android:id="@+id/MyChats"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_gravity="center"
    android:layout_margin="5dp">

    <ImageView
        android:layout_width="50dp"
        android:layout_height="35dp"
        android:layout_gravity="center"
        android:src="@drawable/mychats" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="My chats"
        android:textAlignment="center"
        android:textColor="@android:color/black"
        android:textSize="15sp"
        android:textStyle="bold" />

</LinearLayout>

<LinearLayout
    android:id="@+id/sellBooksBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:orientation="vertical"
    android:layout_margin="5dp">

    <ImageView
        android:layout_width="50dp"
        android:layout_height="35dp"
        android:layout_gravity="center"
        android:src="@android:drawable/ic_menu_add" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="Sell"
        android:textAlignment="center"
        android:textColor="@android:color/black"
```

```
        android:textSize="15sp"
        android:textStyle="bold" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/MyNotifications"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="vertical"
        android:layout_margin="5dp">

        <ImageView
            android:layout_width="50dp"
            android:layout_height="35dp"
            android:layout_gravity="center"
            android:src="@drawable/notifications"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="Notifications"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold" />

    </LinearLayout>
    <LinearLayout
        android:id="@+id/MyBookAds"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="vertical"
        android:layout_margin="5dp">

        <ImageView
            android:layout_width="50dp"
            android:layout_height="35dp"
            android:layout_gravity="center"
            android:src="@drawable/bookpost" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="My Ads"
            android:textAlignment="center"
            android:textColor="@android:color/black"
            android:textSize="15sp"
            android:textStyle="bold"/>

    </LinearLayout>

</LinearLayout>
```

```
</LinearLayout>
```

Fragment_public_institute

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

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewPosts"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>

</RelativeLayout>
```

Fragment_private_institute

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

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recycleViewPosts"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="8dp"/>

</RelativeLayout>
```

Item_communication

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/linear"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:layout_margin="10dp">
    <LinearLayout
        android:paddingBottom="10dp"
        android:gravity="center"
        android:id="@+id/linearDate"
        android:layout_gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <androidx.cardview.widget.CardView
            android:id="@+id/cardDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            app:cardCornerRadius="10dp">
            <TextView
                android:id="@+id/textDate"
                android:layout_marginStart="10dp"
                android:layout_marginEnd="10dp"
```

```
        android:layout_marginTop="5dp"
        android:layout_marginBottom="5dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="11/06/2020">

    </TextView>

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

<TextView
    android:id="@+id/from"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="53dp"
    android:textStyle="italic"
    android:text="Name"
    android:layout_marginLeft="53dp" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/linear2"
    android:layout_gravity="center_vertical"
    android:orientation="horizontal">
    <de.hdodenhof.circleimageview.CircleImageView
        android:layout_width="40dp"
        android:id="@+id/imageView"
        android:layout_height="40dp"
        android:layout_gravity="top"
        android:src="@drawable/profile"/>
    <androidx.cardview.widget.CardView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/cardView"
        android:paddingTop="5dp"
        android:paddingBottom="5dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"
        app:cardCornerRadius="15dp">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="5dp"
            android:layout_marginEnd="5dp"
            android:layout_marginTop="10dp"
            android:visibility="gone"
            android:layout_marginBottom="10dp"
            android:id="@+id/message"
            android:textColor="@color/white"
            android:text="message"/>
        <ImageView
            android:id="@+id/imageMessage"
            android:layout_width="230dp"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_menu_gallery"
            android:visibility="gone" />
    
```

```

        android:scaleType="center" />

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

<TextView
    android:id="@+id/time"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="53dp"
    android:textSize="10dp"
    android:textStyle="italic"
    android:text="12:00AM"
    android:layout_marginLeft="53dp" />

</LinearLayout>
```

Toolbar

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.Toolbar
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/toolbar"
    android:minHeight="?attr/actionBarSize"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:titleTextColor="@android:color/white"
    android:tag="@string/app_name"
    android:background="@color/colorPrimaryDark"
    app:menu="@menu/menu">
</androidx.appcompat.widget.Toolbar>
```

CRAWLER CODE PYTHON

```

from bs4 import BeautifulSoup, SoupStrainer
import requests
import pandas as pd
import urllib.request
import shutil
url = "https://merafuture.pk/PakistaniUniversitiesPakistaniDegrees"
page = requests.get(url)
data = page.text
soup = BeautifulSoup(data, 'html.parser')
images = soup.find_all("a", class_='c-gridpost__image w-inline-block text-center')

image_info = []

for a in images:
    image_tag = a.findChildren("img")
    image_info.append((image_tag[0]["src"], image_tag[0]["alt"]))

df_images = pd.DataFrame({'InstituteImages':image_info})
df_images
info = soup.find_all("div", {'class':'c-gridpost__card'})
institute_info=[]
for i in range(0, len(info)):
    institute_info.append(info[i].get_text())
```

```

df_info = pd.DataFrame({'InstituteInfo':institute_info})
print(df_info)
df_info = df_info.replace(r'\n\n\n\n', '*', regex=True)
df_info= df_info.replace(r'\n\n\n', '*', regex=True)
df_info= df_info.replace(r'\n', '*', regex=True)
df_info
df_info2 = pd.DataFrame(df_info.InstituteInfo.str.split('*').tolist())
'
columns = ['InstituteImages','Institut
eName',
'MajorProgram','Location',
'MajorFields','Type',
'ProgramDuration'])

df_info2['InstituteImages']=image_info
df_info2.to_csv('InstituiteInformation.csv', index=False, encoding='utf
-8')
df_info2

```

Appendix D – Team Members

Jaweria Asif (b. December 15, 1998) is currently enrolled in the Bachelors of Computer Science program at PAF-KIET, majoring in Software engineering currently enrolled in 7th semester. Have furnished my skills at the Android studio using (JAVA),C# and Machine learning using (Python) however, my domain knowledge is more focused on core routing.

Email: Jaweria.asif98@gmail.com

Javeria Hassan (b. July 26, 2000) is currently enrolled in the Bachelors of Computer Science program at PAF-KIET, majoring in Software engineering currently enrolled in 7th semester. Have furnished my skills at the Android studio using (JAVA), C# and Machine learning using (Python) however, my domain knowledge is more focused on core python.

Email: javeria.hassan77@gmail.com

Annexure I – Copy of Approved FYP Proposal

FINAL YEAR PROJECT PROPOSAL [FALL 2020]



Karachi Institute of Economics and Technology
College of Computing and Information Sciences
FINAL YEAR PROJECT – 2020

SEMESTER	FALL 2020	YEAR	202 0-2 021

TITLE OF PROPOSED PROJECT			
A complete solution to a book exchange system with students help			
Project Category (choose one)		<input type="checkbox"/> Product based	<input type="checkbox"/> Research-based
SUPERVISOR INFORMATION			
Supervisor Name:		(leave blank)	Organization/ Designation
Contact No:		(leave blank)	email: (leave blank)
STUDENT(S) INFORMATION			
S#	Student ID		N a m e
1	9517		JAVERIA HASSAN
Contact No:	03042950168	email:	javeria.hassan77@gmail.com
2	9442		JAWERIA ASIF
Contact No:	03323793451	email:	jaweria.asif98@gmail.com
PROJECT AREA/TOOLS			
Tools Required:	Android Studio, SQL Server, Visual studio		
Area/Specialization:	Machine Learning		
SUMMARY OF PROPOSED PROJECT (MAXIMUM 300 WORDS)			
<p>This application aims to support students in a way that it will guide students in which domain the student should continue their further education after matric. After Login, Students can search best college, university, other educational institutes based on their education, academic results, interest, education domain, location and range of fee. By entering correct percentage the student will be able to see the list of colleges and universities catering that particular percentages student. The students will be able to exchange books in low price, from other students by contacting them. Those students who have done their intermediate level and want to search universities according to their range of fees they can afford including their percentages (marks gain in intermediate), ultimately this application will display the list of universities better suited for that particular student. This application will also provide scholarship details and also those institutes which are providing scholarships to the needy students. This application will also recommend courses on behalf of education, skills and interests of students. If a student's wants to go to abroad this application will guide that student as well about which test is needed to be cleared to go to abroad. Students can ask questions using chatting option. This is an AI based system that will do all these recommendations. Initially, this app is just for Karachi only. Using student's queries data, we will implement student's problem analyzer in FYP2. We will apply sentiment analysis on student's book selling reviews. Using book reviews data we will implement book seller recommender.</p>			
PROJECT OBJECTIVE(S)/OUTCOMES			
<p>This app aims to guide those students who are willing and passionate and looking forward towards further education. Using these app students can find career guidance, which courses they should take in their spare timings. This app provides access to learning resources.</p>			

FUNCTIONAL FEATURES

For FYP 1:

1. Login/Sign up.
2. Admin Side.
3. Students Side.
4. Exchange of old books (Students can sell their old books using same app).
5. Book Seller Recommender system.
6. Maintained Data Base.
7. Crawler.
8. Chat Options.
9. Email Notification.
10. Mobile Number Verification. 
11. Email Verification.
12. User data Management.
13. Students Discussion Forum.

For FYP 2:

1. Student Problem Analyzer using student queries data.
2. Sentiment Analysis.
3. AI based recommender system.

For Proposal Defense Purpose

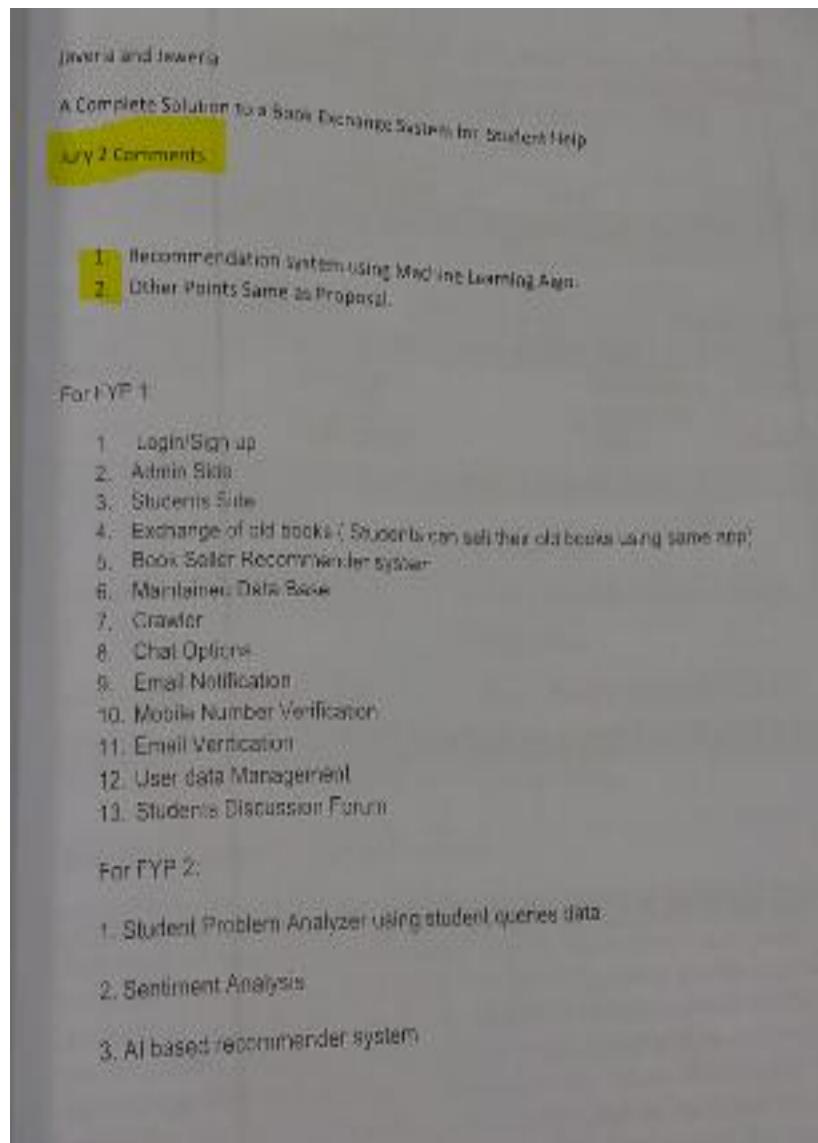
PROPOSED ADVISORY COMMITTEE		
S#	Faculty Name	Signature
1		
2		
3		
4		

FYP Committee

FYP COMMITTEE			
S#	Member(s) Name	Designation	Signature

1	Syed Nabeel Ali (FYP Coordinator)	Lecturer	
2	Mr. Kashif Bashir (Head of the Department)	Asst. Professor	

APPROVED PROPOSAL WITH JURY COMMENTS



Need Approval for the changes in Functional Requirements.

Inbox x



jaweria asif <jaweria.asif98@gmail.com>
to usman, usmanarab, ehtesham_khan, s.nabeel ▾

Tue, Sep 15, 1:38 PM



Assalam-o-Alikum Sir,
Well, we are doing FYP and after the first meeting we realized the following things.

Approval #01:

We should go with Firebase instead of SQL Server but in FYP proposal we have mentioned only SQL Server. So, we want you to kindly give us both options SQL Server/ Firebase.

If you allow us to use Firebase then we will use Google-Firebase API and in this API we will use storage, real-time database and authentication

Approval #02:

Also, we forgot to mention python in FYP proposal as we are going to use python models for applying data analytics on the application. So, this is a humble request to kindly approve python as well.

Approval #03:

We want you to kindly remove feature#10 which is Mobile Number Verification.

I have attached the FYP proposal for your reference.

We will be very grateful to you.

Regards,
Jaweria Asif(9442)
Javeria Hassan(9517)

Activate Windows
Go to Settings to activate Win

usman khan <usman@pafkiet.edu.pk>
to usmanarab, ehtesham_khan, s.nabeel, me ▾

Tue, Sep 15, 1:45 PM



Wsalam,
Ok approved 1 and 3.
I need more details for Approval #02 like which python model you are talking about and which Data analytic application?

Kind Regards

jaweria asif <jaweria.asif98@gmail.com>
to usman ▾

Tue, Sep 15, 1:52 PM



Thanks sir,
For Approval #02:
For recommending book we will use Collaborative filtering model
For Sentiment Analysis we use Linear Regression, Naive Bayes, SVM
For Crawling we will use Selenium, Beautiful Soup, Pandas, Numpy, Browser will be Google chrome
and for this we want to use python as we have learnt these models in Machine learning.

...

usman khan
Ok approved. Attached a copy of this email with your proposal.

Sep 15, 2020, 2:01 PM



jaweria asif <jaweria.asif98@gmail.com>
to usman ▾

Sep 15, 2020, 2:06 PM



Thanks a lot sir.

...

Summary Table

Title of Project	Book Exchange Application
Nature of End Product	Mobile App
Programming Environment	64-bit windows 10
Programming Languages	Java, Python
Software Tools	Android Studio, Firebase, Jupyter Note Book
Team Members	Javeria Hassan (9517) Jaweria Asif (9442)

Accepted / Rejected: _____

Total Marks:

Committee Review:

Committee Signatures:

Annexure I – Use Case Specifications

Use Case ID:	01
Use Case Name:	Update Profile
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User/ Admin
Description:	Each user whether acting as a buyer, seller or the admin has a personal account that he/she maintains and updates regarding their personal information.
Preconditions:	<ul style="list-style-type: none"> 1) The user/Admin is part of the system 2) The user has a valid unique id generated by the database and he/ she should be a registered user.
Post conditions:	<ul style="list-style-type: none"> 1) An updated profile would be created that can be accessed and viewed by the admin and the particular actor updating their profile.
Normal Flow:	<ul style="list-style-type: none"> 1) The user/admin logs in to his/her account. 2) The user goes to the menu item/ card view 3) The user updates personal profile 4) The user saves these changes and they are now stored in the database 5) The user logs out
Alternative Flows:	<ul style="list-style-type: none"> 1) The admin logs in 2) The admin exercises his right to change profile information for users.
Exceptions:	<ul style="list-style-type: none"> 1) There is no user account maintained. 2) The user seeks to change information only the admin can change
Includes:	Login
Priority:	Medium
Frequency of Use:	Depending on the number of users/admin it could range from a couple to hundred times throughout a single month.
Special Requirements:	<ul style="list-style-type: none"> 1) No field can be left empty 2) Data type should match the one given in fields
Assumptions:	All actors would upload correct synchronized and valid information regarding their profile.
Notes and Issues:	The profile must be updated within time to ensure the system has relevant and up-to-date information.

Use Case ID:	02
Use Case Name:	User upload post for selling their old books
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	When user intended to post an ad of their old books for selling through the application. Firstly, it will retrieve towards the admin panel and when admin accepted the post it will retrieve on the timeline and available for the buyers.
Preconditions:	1) The user is part of the system 2) The user has a valid unique id generated by the database and he/ she should be a registered user.
Post conditions:	1) Book status should be approved by admin
Normal Flow:	1) The user logs in to its account 2) Select the option of book-exchange 3) Then user will navigate to the timeline showing recent ads of book posts, On the bottom of the time line User will select add. 4) When user will select post add its image gallery will be open through which the user will select the image of the book. 5) Then the user will fill the description form and click the upload button 6) Then the user will wait for the book status transforms from “not approved” to “approved” by the admin.
Alternative Flows:	None
Exceptions:	1) The user should be authentic and book should be available
Includes:	None
Priority:	High
Frequency of Use:	One post each day by the user will be accepted
Special Requirements:	Book status should be approved before the buyer approach the seller for buying the book.
Assumptions:	None
Notes and Issues:	Once the result is sealed it cannot be changed unless examination allows it

Use Case ID:	03
Use Case Name:	User Setting a query in discussion forum
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	When user suffered some problem related to studies or coding they discuss their problem through discussion forum, Any user set a problem statement and any user who is able to resolve the problem statement can answer the stated query.
Preconditions:	1) The user is part of the system 2) The user has a valid unique id generated by the database and he/ she should be a registered user.
Post conditions:	
Normal Flow:	1) User logs in 2) Select the feature discussion forum 3) Write problem statement as text message in discussion forum 4) Attach image if necessary 5) Wait for the replies
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Until the user get a satisfactory answer
Special Requirements:	The user can attach files and can provide links
Assumptions:	The material posted is decipherable, relevant and in line with the system so it could be viewed and accessed
Notes and Issues:	Everybody should individually check whether a query has been set

Use Case ID:	04
Use Case Name:	User Viewing Educational Institute Data of universities
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	Any user is eligible for viewing data of several different universities of Karachi and can gather useful details about the universities for getting admission for further education
Preconditions:	1) The user logs in
Post conditions:	None
Normal Flow:) User logs in 2) Select the feature educational institute 3) Can see the recent data coming from crawler on the time related to the universities of Karachi.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	Low
Frequency of Use:	Any number of times
Special Requirements:	None
Assumptions:	User belongs to Karachi
Notes and Issues:	Can also be implemented for the rest of the cities

Use Case ID:	05
Use Case Name:	User Viewing Thought of the day
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	Any registered user is eligible for watching the images retrieving on the timeline which enhance motivation level in the users.
Preconditions:	1) The user logs in
Post conditions:	The admin should store the images in database from its panel.
Normal Flow:	<ul style="list-style-type: none">) User logs in 2) Select the feature thought of the day 3) Can see the recent images retrieving on the timeline uploaded by admin of the application.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	Low
Frequency of Use:	Any number of times
Special Requirements:	The admin should store the images in database from its panel before the user open this option.
Assumptions:	The admin had stored the images in database from its panel.
Notes and Issues:	None

Use Case ID:	06
Use Case Name:	User Verification
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User/System
Description:	Any user who wants to use this application must first login to become an authentic user of this application
Preconditions:	1) Signup
Post conditions:	Forget password Remember my password and ID
Normal Flow:	1) User start the application 2) If user is not registered 3) User will be signup first 4) Then user will login 5) Then system will authenticate user
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Any number of time
Special Requirements:	Signup
Assumptions:	User will provide authentic data
Notes and Issues:	None

Use Case ID:	07
Use Case Name:	When book status become “ not available/sold out”
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	When a book user has posted a book ad and any user of the application brought that book from that particular user. So that particular user will edit his/her post as book status sold out/not available, so that the rest of the users who need the same book would be compensated.
Preconditions:	The user logs in The user must has given book ad before updating the book status
Post conditions:	None
Normal Flow:	1) User logs in 2) Select the feature book exchange 3) Click on his/her ad. 4) Update ad 5) Book sold out
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	onetime
Special Requirements:	User must has given the ad
Assumptions:	User must has given the ad
Notes and Issues:	None

Use Case ID:	08
Use Case Name:	Book Recommendation
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	When a user repeatedly search for a specific type of book then application will start recommending the user the books related to that specific category which user has searched for.
Preconditions:	The user logs in The user must have search for a specific category book
Post conditions:	None
Normal Flow:	1) User logs in 2) Select the feature book exchange 3) Click on recommendations
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Any number of time
Special Requirements:	User must have search for a specific category book
Assumptions:	User search for a book
Notes and Issues:	None

Use Case ID:	09
Use Case Name:	Chat against a book ad
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	When a specific user posted an ad for selling one of their old books then any number of users can chat with that particular user (seller) in order to purchase that particular book. Then, the user (seller) will decide whom he want to sell his/her book.
Preconditions:	All users must logs in An ad must be posted against several chats
Post conditions:	None
Normal Flow:	1) User logs in 2) Select the feature book exchange 3) Click on chats
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Any number of time
Special Requirements:	User must have given a post for selling an old book
Assumptions:	User has given the ad
Notes and Issues:	None

Use Case ID:	10
Use Case Name:	User can see his add list on timeline
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	User
Description:	A user can see only his/her posted ads retrieving on the timeline
Preconditions:	1) user must logs in An ad must be posted before watching my ads otherwise it will be empty.
Post conditions:	User can edit his/her posted ads
Normal Flow:	1) User logs in 2) Select the feature book exchange 3) Click on my posted ads
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Any number of time
Special Requirements:	User must have given a post for selling an old book
Assumptions:	User has given the ad
Notes and Issues:	None

Use Case ID:	11
Use Case Name:	Admin upload thought of the day images
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin /System
Description:	Admin is responsible for uploading the images of thought of the day which motivates the users of the application
Preconditions:	Admin must logs in.
Post conditions:	None
Normal Flow:	<ul style="list-style-type: none"> 1) Admin logs in 2) Select the feature Add thought of the day 3) Select on upload image 4) then add desired image from gallery 5) Image uploaded successful
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Any number of time
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	12
Use Case Name:	Admin add book category
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System
Description:	Admin is responsible for adding new book categories in firebase
Preconditions:	Admin must logs in.
Post conditions:	None
Normal Flow:	1) Admin logs in 2) Select the feature Add book category 3) Select add book category 4) Save
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Depends on book categories
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	13
Use Case Name:	Admin View Data of Educational Institutes
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System
Description:	Admin can view data which is coming from crawler by crawling several different website of different universities
Preconditions:	Admin must logs in.
Post conditions:	None
Normal Flow:	1) Admin logs in 2) Select the feature Educational Institute Data 3) View data
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Depends on Admin
Special Requirements:	Crawler must be working before admin use this option
Assumptions:	None
Notes and Issues:	None

Use Case ID:	14
Use Case Name:	Admin View List of users
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System
Description:	Admin can view list of users which are currently using this application
Preconditions:	Admin must logs in.
Post conditions:	None
Normal Flow:	1) Admin logs in 2) Select the feature View users 3) View lists
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Depends on Admin
Special Requirements:	User must be registered in the application
Assumptions:	None
Notes and Issues:	None

Use Case ID:	15
Use Case Name:	Admin View Recommended Data
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System
Description:	Admin can view recommended data which is coming from Models designed by python set in the firebase as Json file.
Preconditions:	Admin must logs in.
Post conditions:	User must Ad book post
Normal Flow:	1) Admin logs in 2) Select the feature View recommended data 3) View data
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Depends on Admin
Special Requirements:	User must using the application
Assumptions:	None
Notes and Issues:	None

Use Case ID:	16
Use Case Name:	Admin can download CSV data
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System
Description:	Admin can download CSV data coming from crawler
Preconditions:	Admin must logs in.
Post conditions:	None
Normal Flow:	<ul style="list-style-type: none"> 1) Admin logs in 2) Select the feature CSV data 3) View data 4) Download data if needed
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	Depends on Admin
Special Requirements:	Crawler must be working before admin use this option
Assumptions:	None
Notes and Issues:	None

Use Case ID:	17
Use Case Name:	Admin Accept post generated by User
Created By:	Jaweria Asif and Javeria Hassan
Date Created:	1-1-2021
Actors:	Admin/System/ user
Description:	Admin is responsible for making post status accepted or rejected and when the status of that particular post become accepted user can buy that particular book which is available in the post.
Preconditions:	Admin must logs in. User must have given post for approval
Post conditions:	None
Normal Flow:	1) Admin logs in 2) Select the feature Book posts request 3) Accept/ Decline post
Alternative Flows:	None
Exceptions:	None
Includes:	None
Priority:	High
Frequency of Use:	One time
Special Requirements:	User must have given post for approval
Assumptions:	None
Notes and Issues:	none