IT2010 – Mobile Application Development

BSc (Hons) in Information Technology

2nd Year

Faculty of Computing

SLIIT

2023 – Assessment 02

Description	Student ID	Name
Group Leader	IT21156960	Kodithuwakku C.K
Member 2	IT21164330	Arandara S.D.
Member 3	IT21166174	Navindi R.L.S
Member 4	IT21184444	Sumanasekara W.H.U.

Application Topic	Online education application
Group Name	**Code warriors**

Project Declaration

We, the members of Code Warriors, hereby declare that our group project is entirely authentic and original. We have conducted thorough research and analysis to ensure that our work is not plagiarized or copied from any other sources.

We have followed all guidelines provided by our LIC and have complied with all ethical and academic standards.

We take full responsibility for the authenticity of our work and understand the implications of academic dishonesty. We have worked collaboratively to produce this project, and each member has contributed to the best of their abilities.

We hereby affirm that our project represents our honest effort and commitment to academic integrity, and we take pride in presenting it as our own.

Group Leader (Signature)
<<Group Leader Name>>

Contents

Introduction

As a result of current unstable situation of politics and economy, Sri Lanka's education sector is experiencing significant issues.

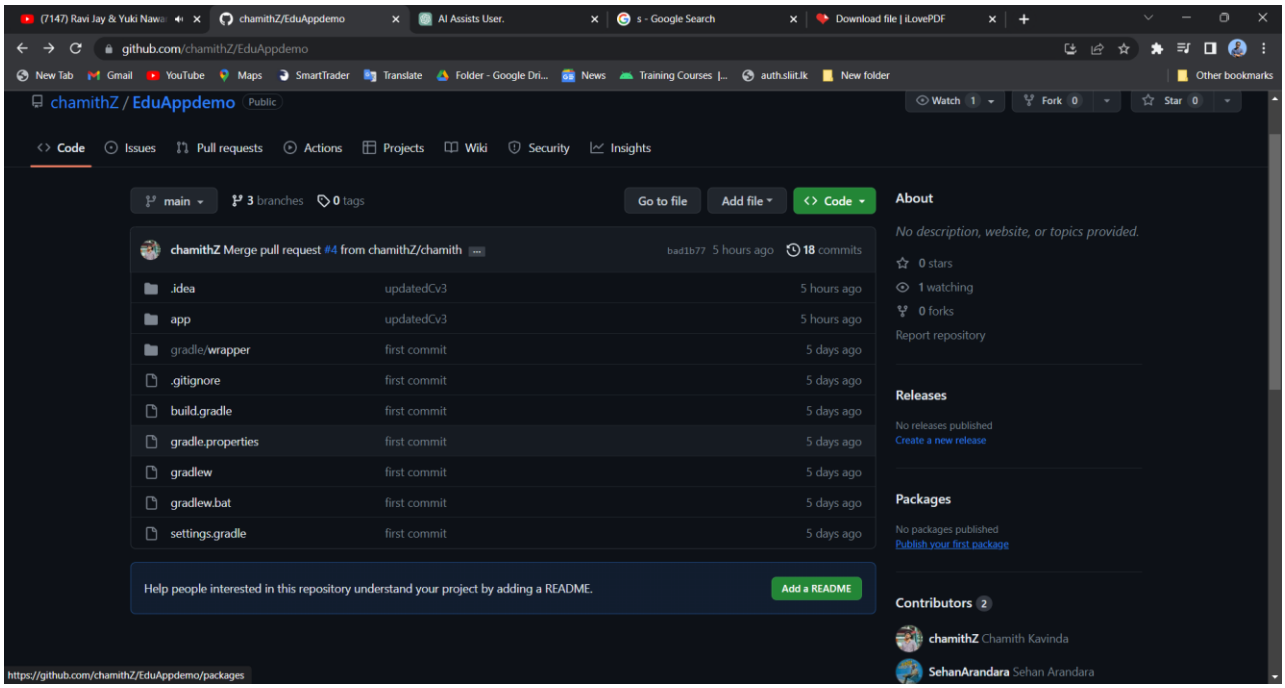
Many colleges and institutions struggle to provide basic services and facilities to the students due to lack of funding. As a result, education has declined in quality, and many pupils are finding it difficult to keep up with their studies. So, to overcome this problem, our developing team has come up with the idea of an app called “EduX” to help the students in our country.

With the help of various innovative tools, the "EduX" app provides students with all the educational support they need to advance their knowledge. This mobile application involves three principal parties. They are students, teachers, and donors. Students can enroll for courses and quizzes. The service that the teachers are offering is free, but if any donors would like to contribute money for it, they can do so through this app. Donating money isn’t compulsory here. Donors who want to give books for the benefit of students can do so through this approach. Users now have the option to create and publish blog posts directly from their mobile device. We have made it easy for users to write and design their blog entries, including photographs and videos, and to preview their work before publishing.

Our team's ultimate objective in creating this educational application was to give users a fun and engaging way to learn even in the midst of a serious economic crisis in the nation, as well as to give them hope for their future aspirations.

Git Repository:

https://github.com/chamithZ/EduAppdemo



Contribution

1. IT21156960 (Kodithuwakku C.K)- Class management and Progress tracker

Class management and Progress tracker

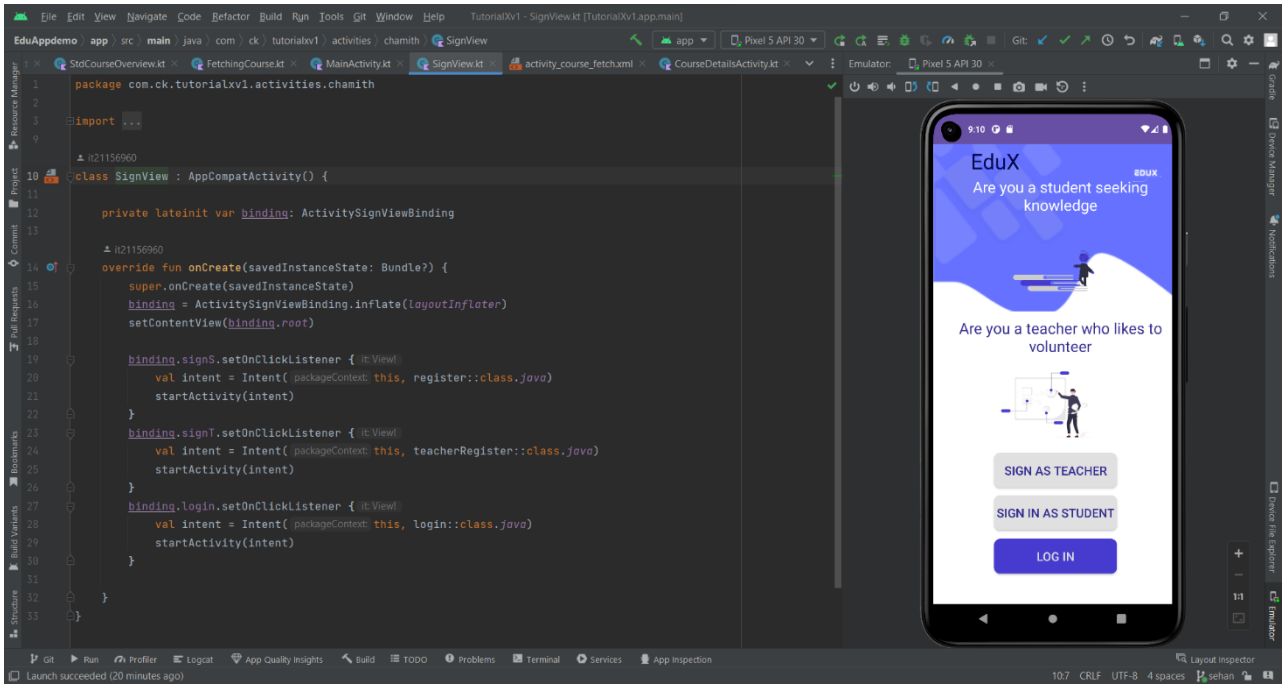
As an individual responsible for managing courses and users in this project, I was tasked with providing a user-friendly interface for students to register and view courses. I was also responsible for enabling students to enroll in courses and view their enrolled courses in a dedicated "My Courses" section. If a student wished to unenroll from a course, they could do so easily.

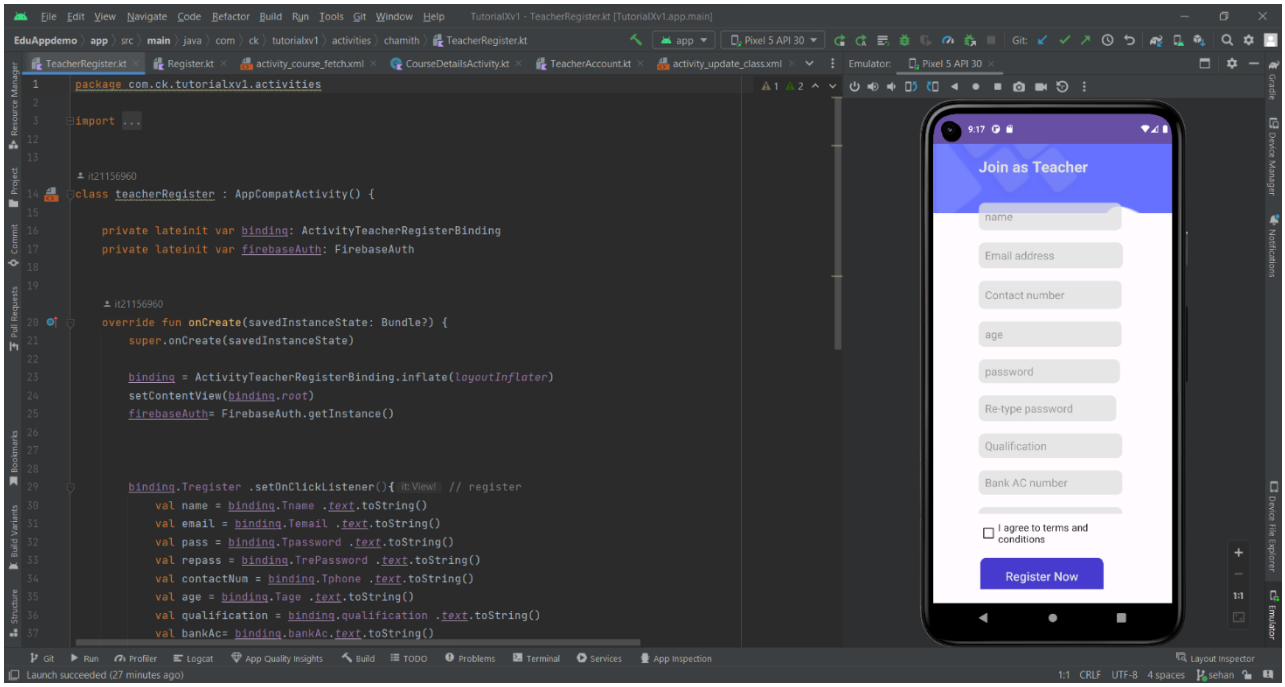
The enrollment feature was designed to incentivize students to continue learning by awarding points for each course they enrolled in. The number of points awarded depended on the user's level. These points were then used to calculate the user's progress, which was displayed in their profile. The progress tracking feature was effective in motivating students to continue learning and achieving their goals.

In addition to managing students, I was responsible for enabling teachers to register and add classes to the system. The system provided teachers with an easy-to-use interface to add classes with subject, grade, time, and date details. Teachers could also view their courses and update or delete them if required.

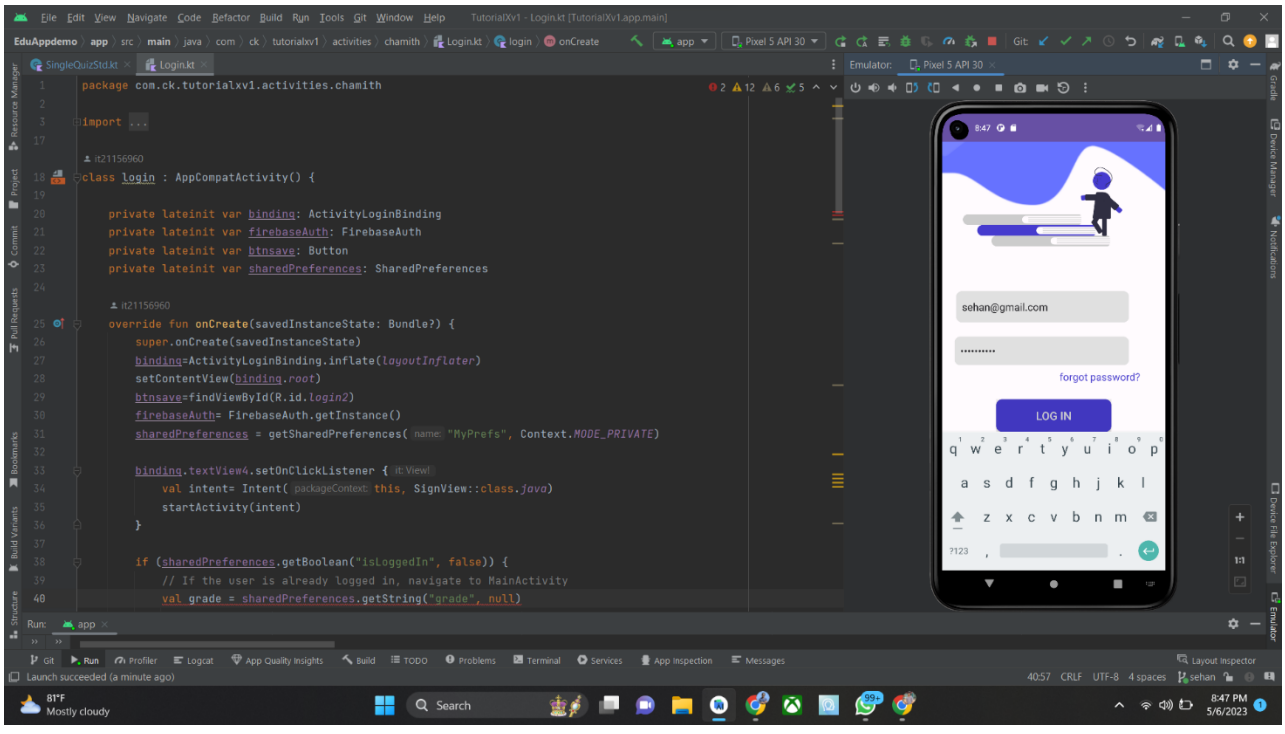
The course management features proved to be valuable additions to the system. They allowed students to easily find and enroll in courses and enabled teachers to manage their classes effectively. Additionally, the points-based enrollment system incentivized students to continue learning and improved their overall performance. The progress tracking feature provided students with valuable feedback on their progress and motivated them to continue learning.

Signup :

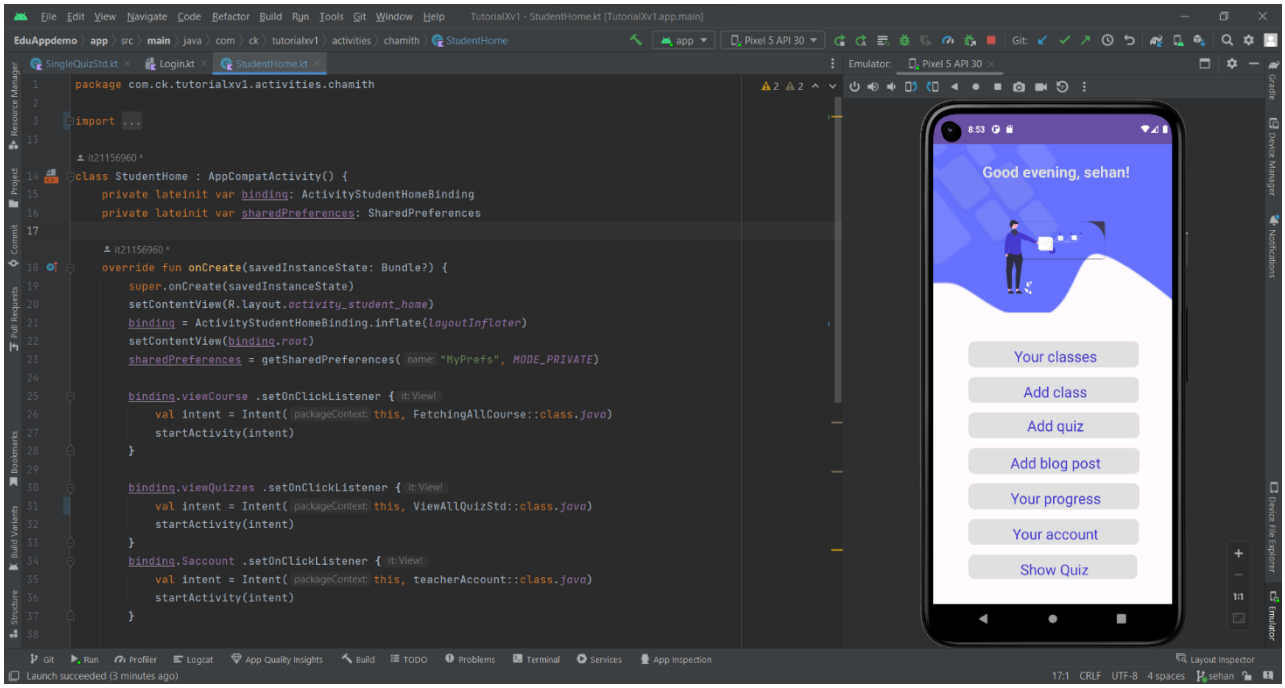


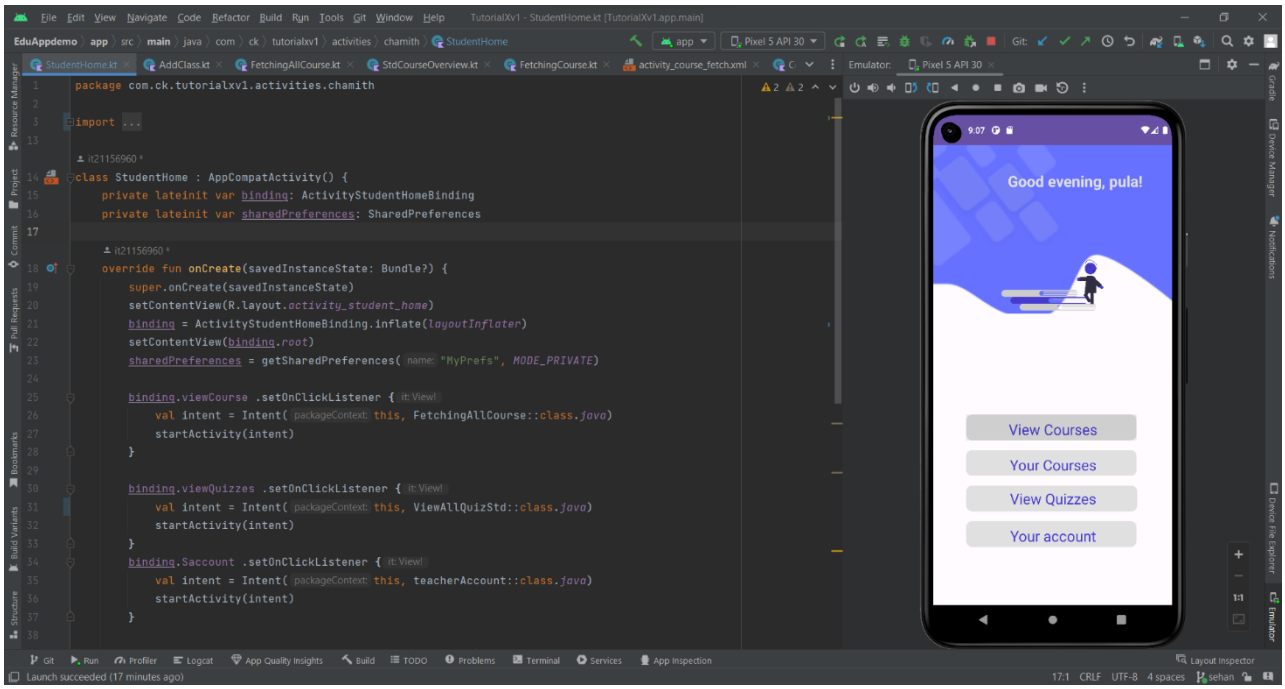

package com.ck.tutorialxv1.activities
import ...
class teacherRegister : AppCompatActivity() {
private lateinit var binding: ActivityTeacherRegisterBinding
private lateinit var firebaseAuth: FirebaseAuth
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
binding = ActivityTeacherRegisterBinding.inflate(layoutInflater)
setContentView(binding.root)
firebaseAuth= FirebaseAuth.getInstance()
binding.Tregister .setOnClickListener(){ // register
val name = binding.Tname .text.toString()
val email = binding.Temail .text.toString()
val pass = binding.Tpassword .text.toString()
val repass = binding.TrePassword .text.toString()
val contactNum = binding.Tphone .text.toString()
val age = binding.Tage .text.toString()
val qualification = binding.qualification .text.toString()
val bankAc= binding.bankAc.text.toString()
Join as Teacher
name
Email address
Contact number
age
password
Re-type password
Qualification
Bank AC number
I agree to terms and conditions
Register Now

Login :


package com.ck.tutorialxv1.activities.chamith
import ...
class login : AppCompatActivity() {
private lateinit var binding: ActivityLoginBinding
private lateinit var firebaseAuth: FirebaseAuth
private lateinit var btnsave: Button
private lateinit var sharedPreferences: SharedPreferences
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
binding=ActivityLoginBinding.inflate(layoutInflater)
setContentView(binding.root)
btnsave=findViewById(R.id.login2)
firebaseAuth= FirebaseAuth.getInstance()
sharedPreferences = getSharedPreferences(name: "MyPrefs", Context.MODE_PRIVATE)
binding.textView4.setOnClickListener { it: View!
val intent= Intent(packageContext: this, SignView::class.java)
startActivity(intent)
}
if (sharedPreferences.getBoolean("isLoggedIn", false)) {
// If the user is already logged in, navigate to MainActivity
val grade = sharedPreferences.getString("grade", null)
sehan@gmail.com
forgot password?
LOG IN

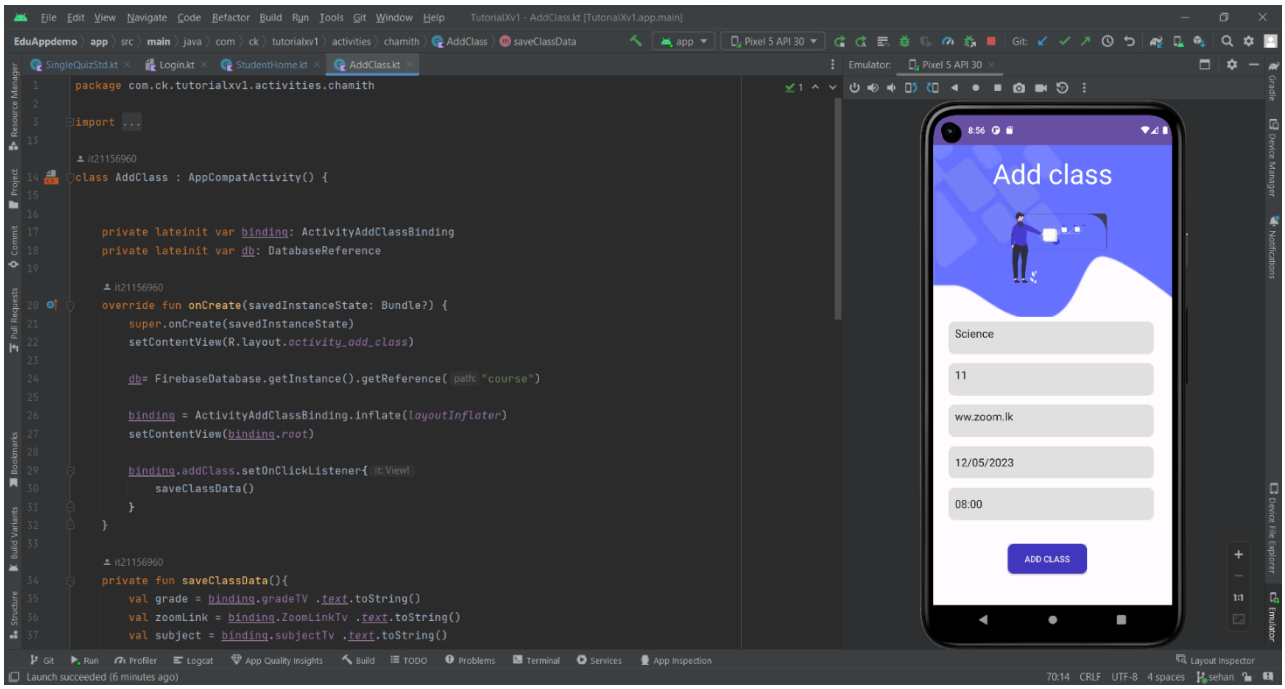
Home Page


Good evening, sehan!
Your classes
Add class
Add quiz
Add blog post
Your progress
Your account
Show Quiz

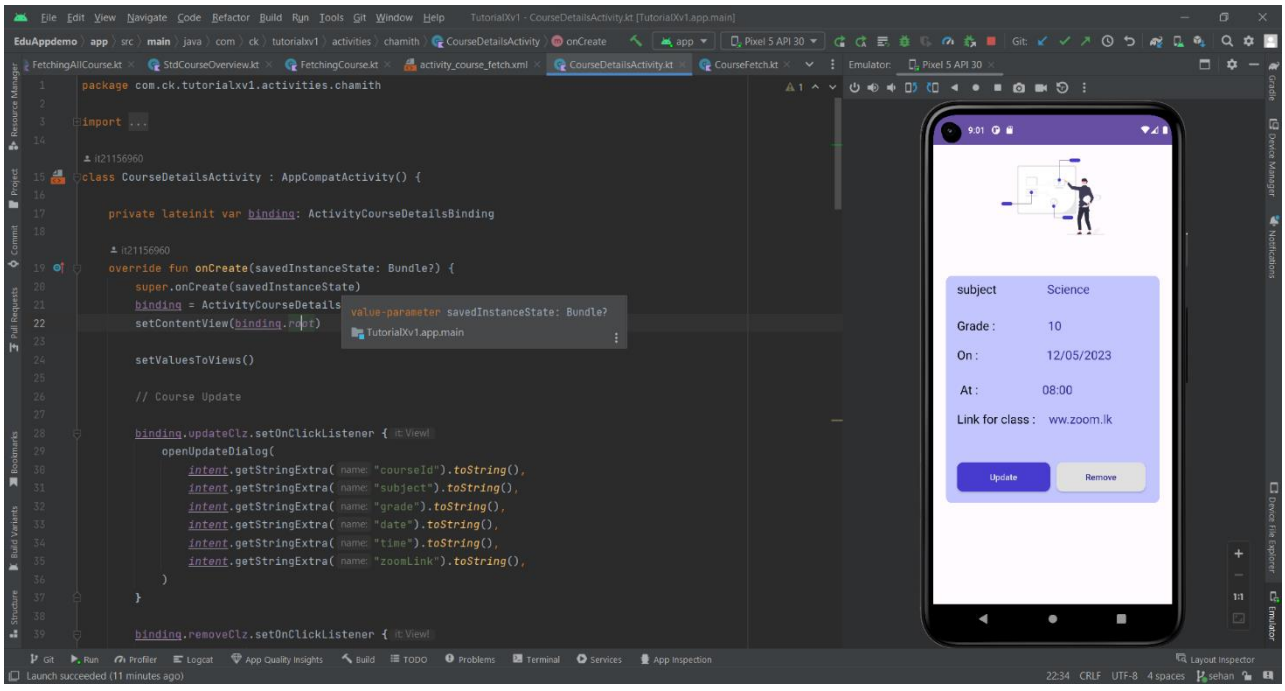

Good evening, pula!
View Courses
Your Courses
View Quizzes
Your account

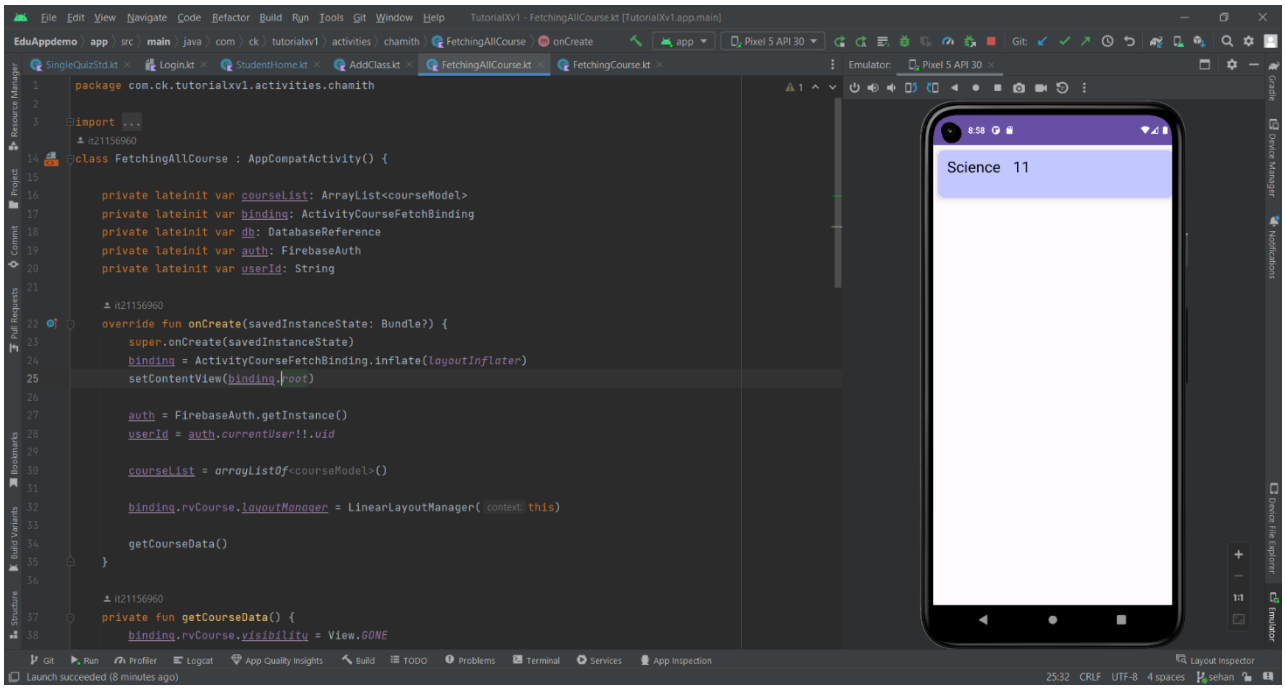
CRUD OPERATIONS

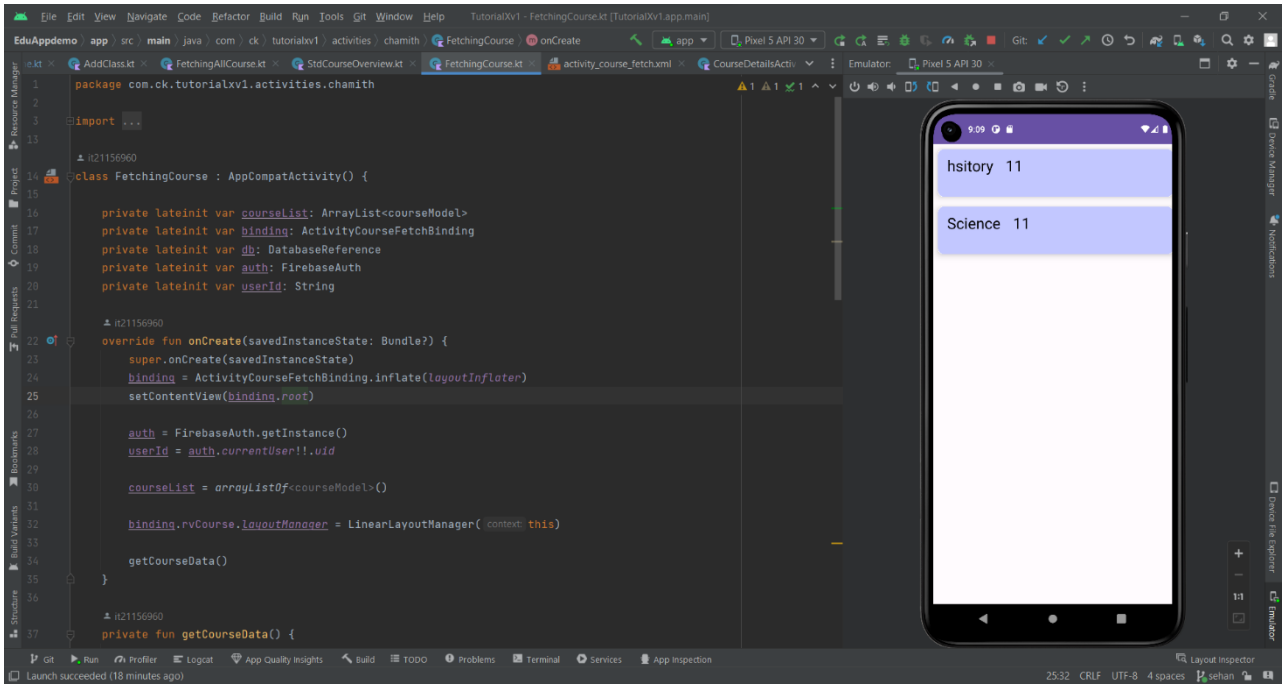
1. Create Operation



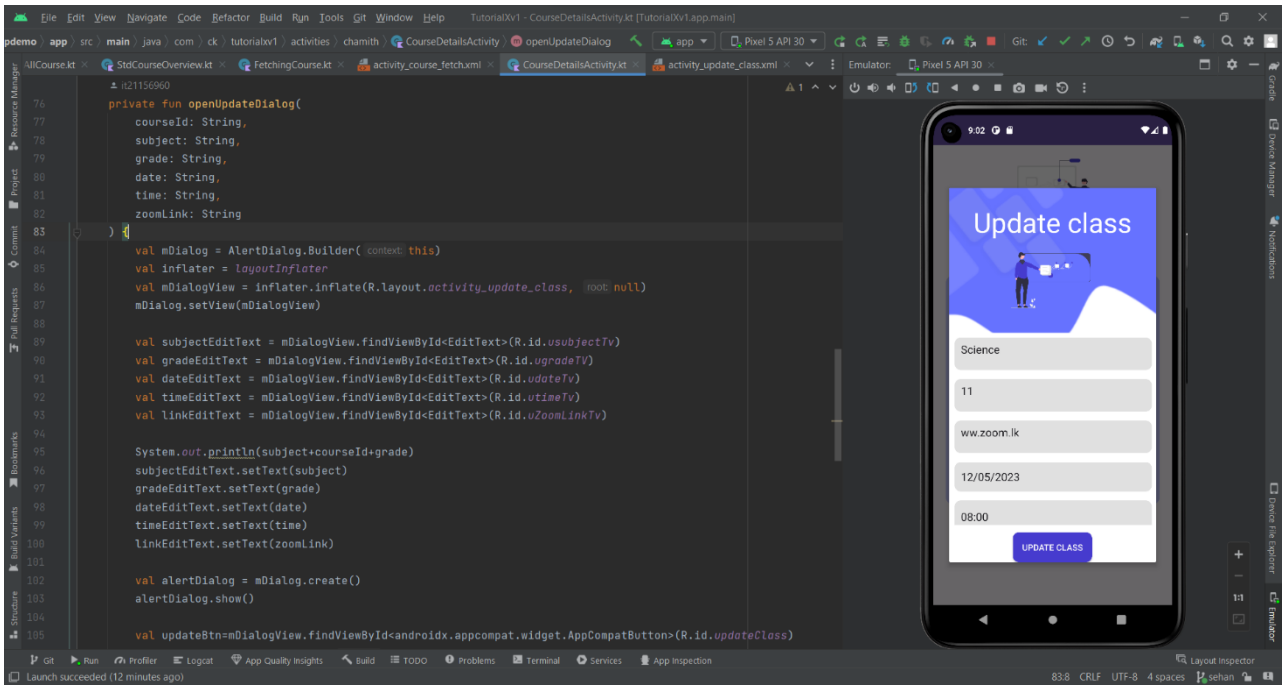
2. Read Operation



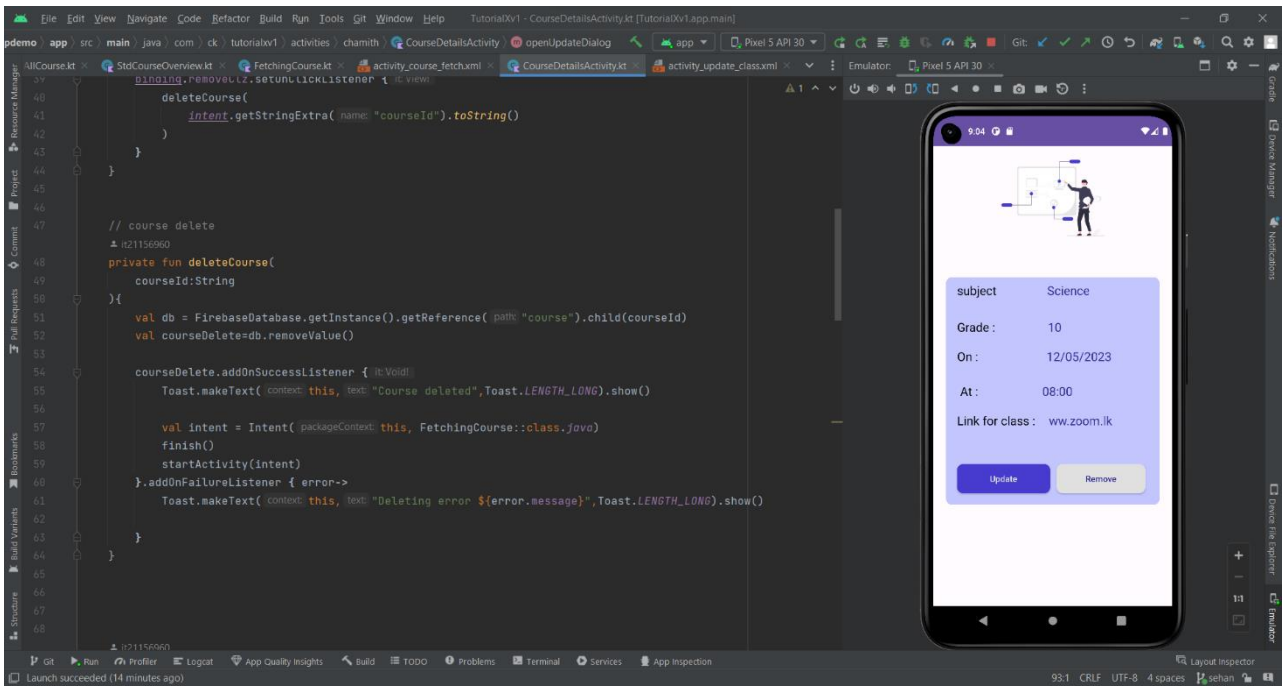

Science 11


hsitory 11
Science 11

3. Edit Operation



4. Delete Operation

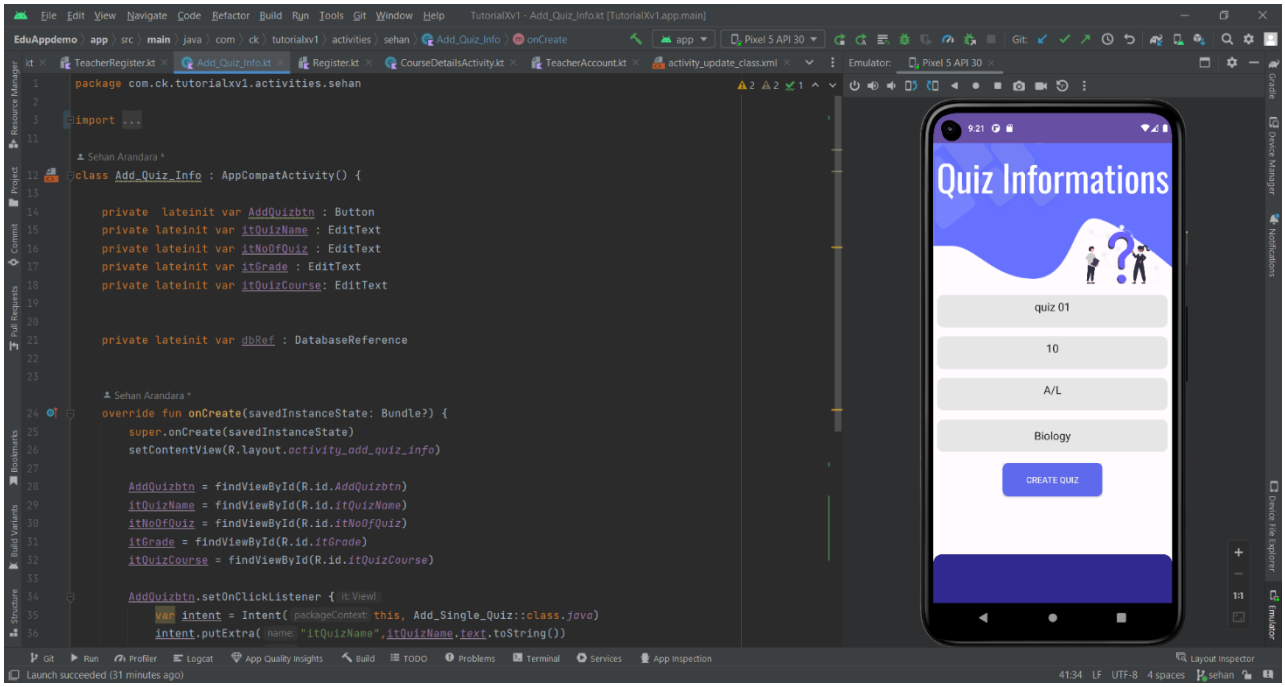


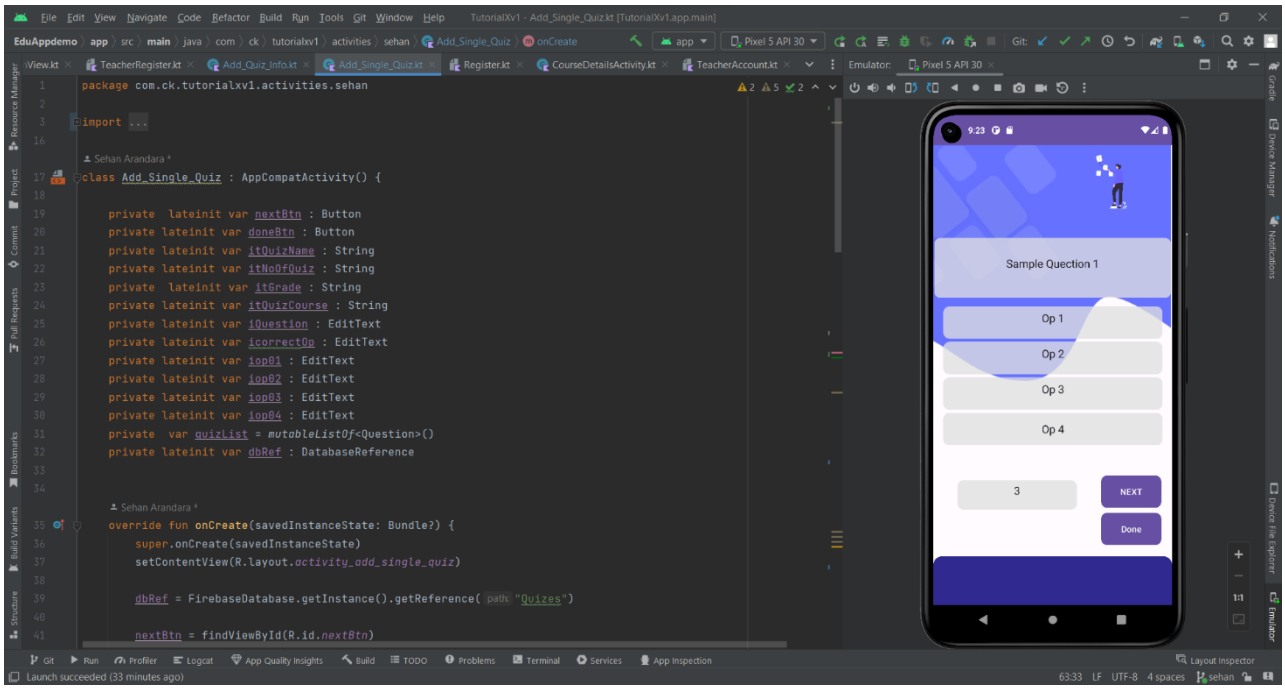
2. IT21164330 (Arandara S.D.) - Quiz management

I am the one who responsible for implement the quiz section in our app . In Our EduX app aims to provide free education to student , so that we are implementing quizzes . It is a vey good way to test their knowledge in various subjects . Teacher can create quizzes by providing the necessary information about the quiz . teacher can edit or delete the quizzes as they want. Both teachers and student can view the quizzes and student can do the quizzes . I will implement some function to calculate the marks and the grade of the student after the quiz finished.

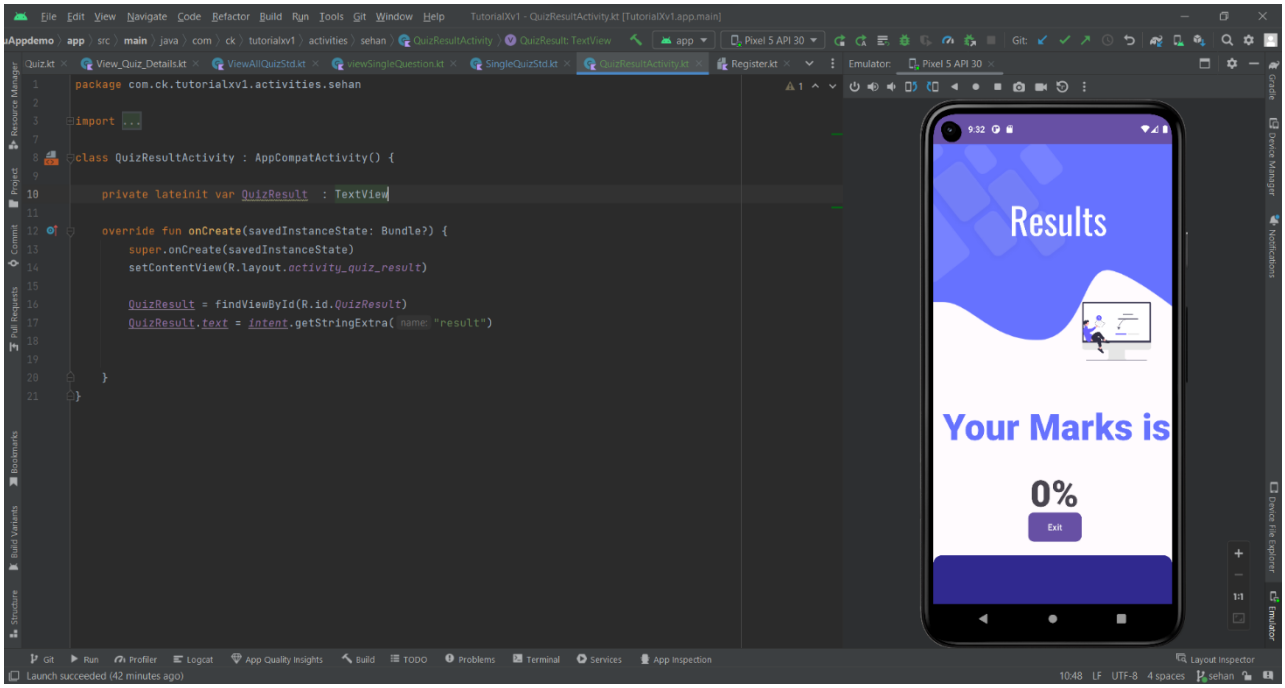
CRUD OPERATIONS

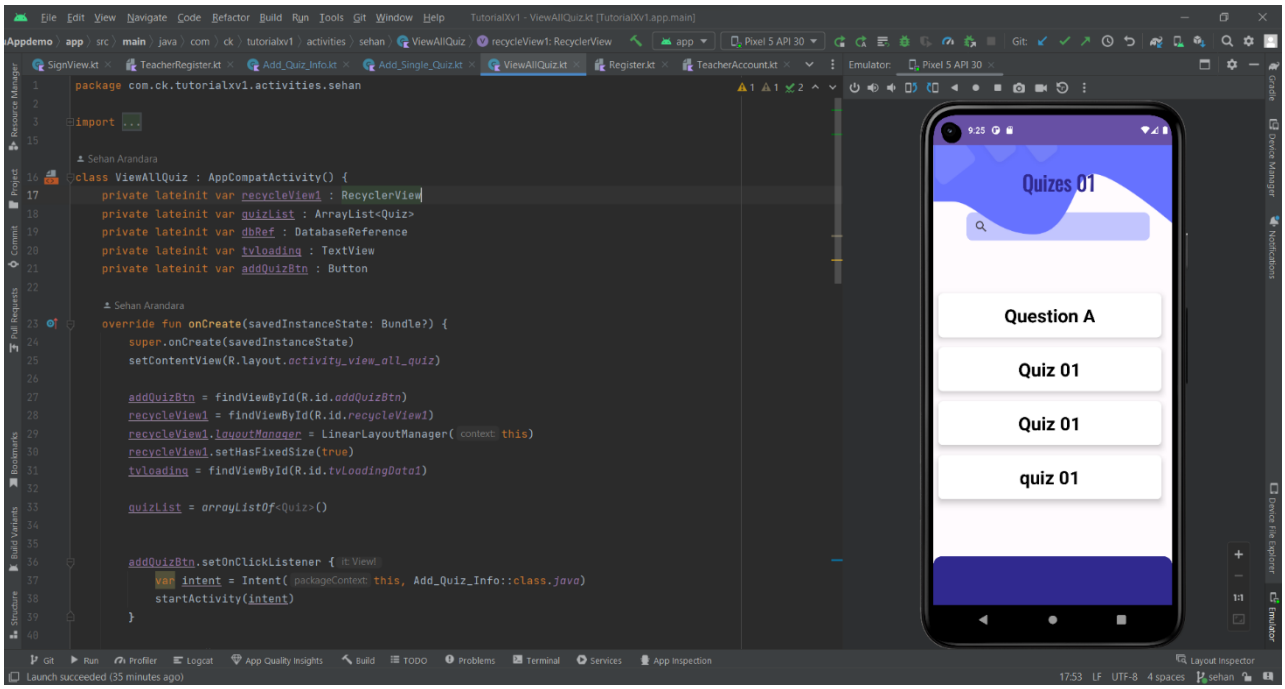
1. Create Operation

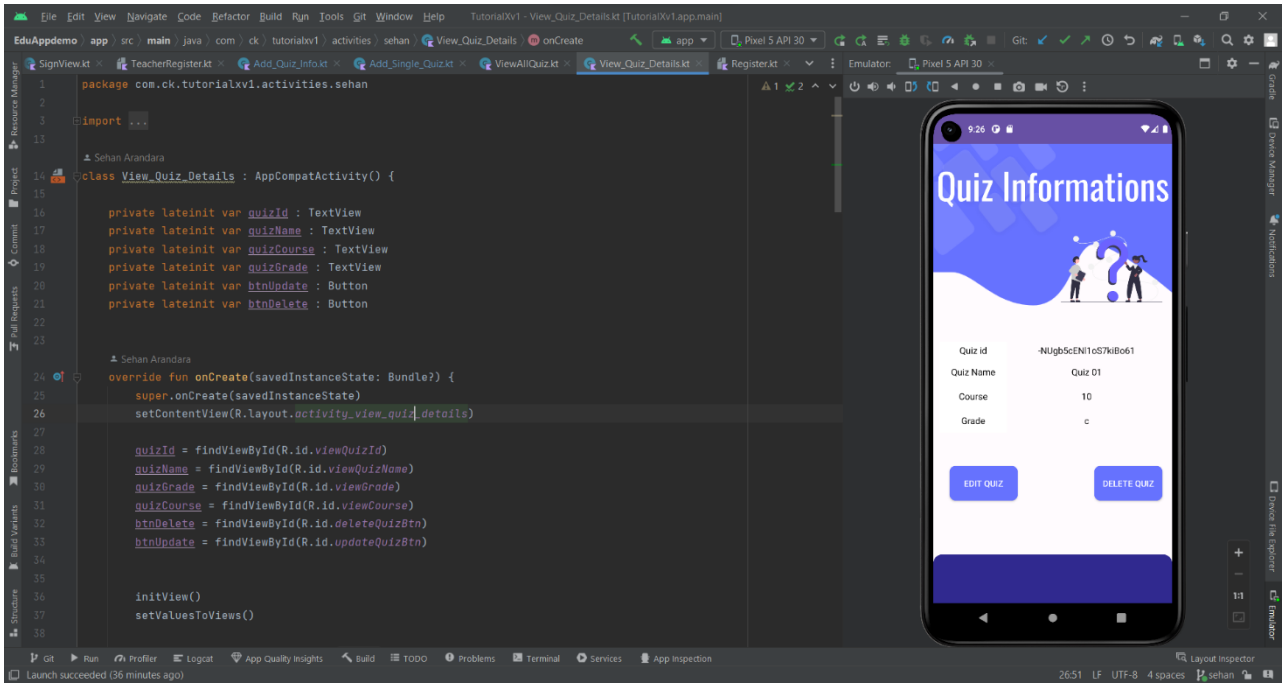


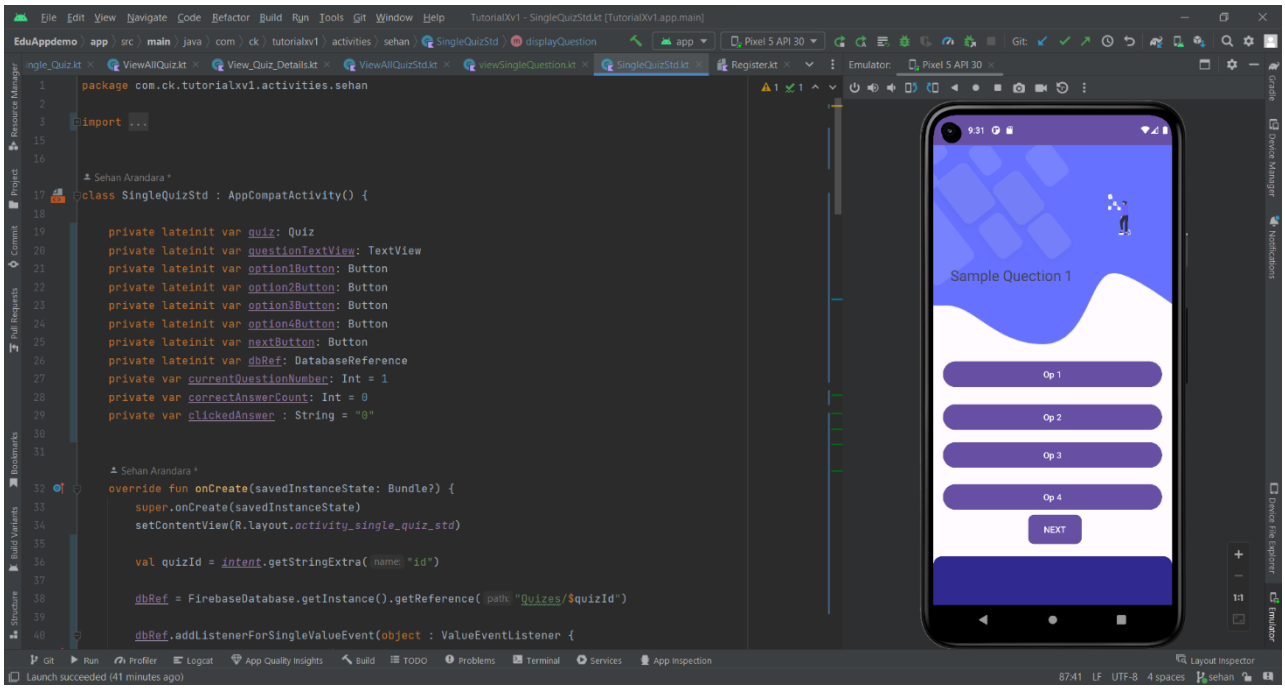


2. Read Operation

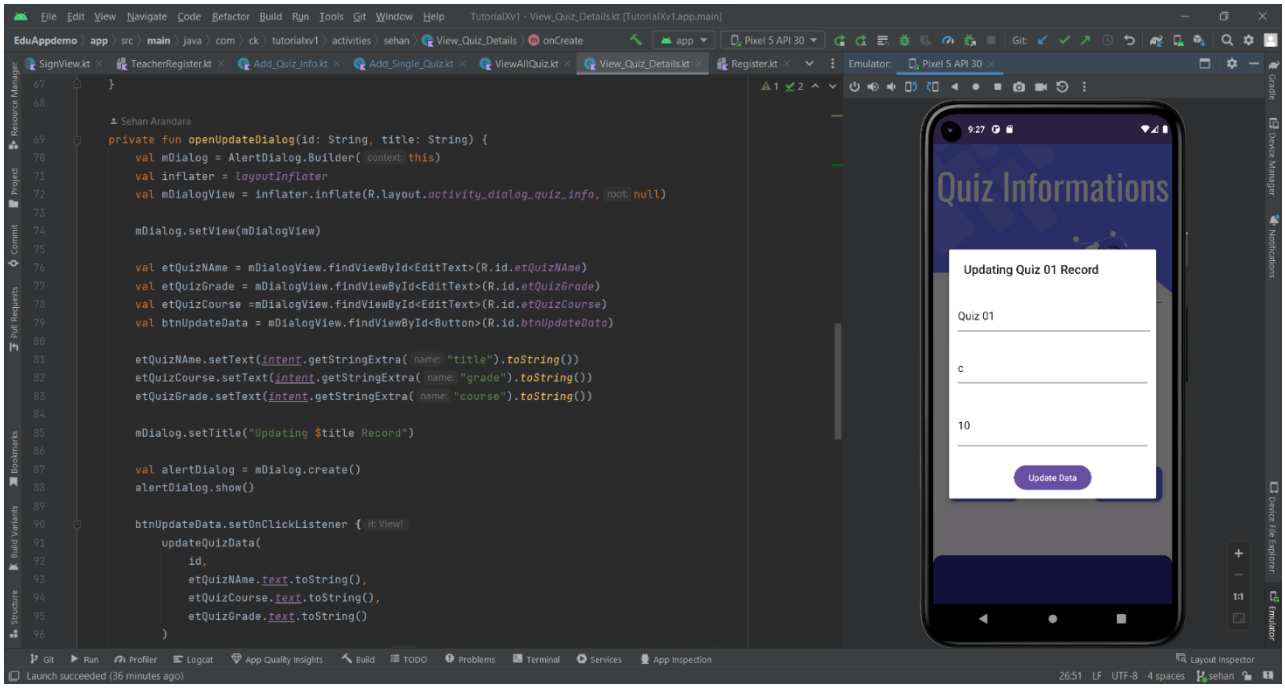



package com.ck.tutorialxv1.activities.sehan
import ...
Sehan Arandara
class ViewAllQuiz : AppCompatActivity() {
private lateinit var recycleView1 : RecyclerView
private lateinit var quizList : ArrayList<Quiz>
private lateinit var dbRef : DatabaseReference
private lateinit var tvloading : TextView
private lateinit var addQuizBtn : Button
Sehan Arandara
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContentView(R.layout.activity_view_all_quiz)
addQuizBtn = findViewById(R.id.addQuizBtn)
recycleView1 = findViewById(R.id.recycleView1)
recycleView1.layoutManager = LinearLayoutManager(context: this)
recycleView1.setHasFixedSize(true)
tvloading = findViewById(R.id.tvLoadingData1)
quizList = arrayListOf<Quiz>()
addQuizBtn.setOnClickListener { it: View!
var intent = Intent(packageContext: this, Add_Quiz_Info::class.java)
startActivity(intent)
}
Quizes 01
Question A
Quiz 01
Quiz 01
quiz 01

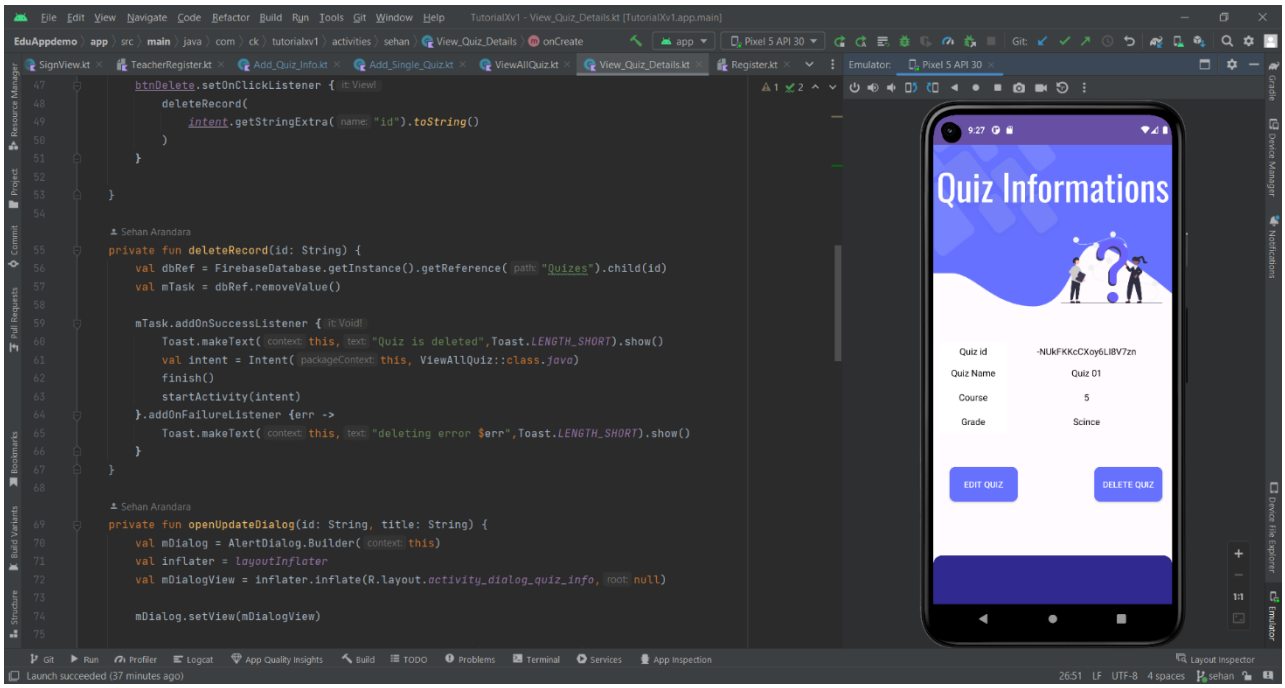

package com.ck.tutorialxv1.activities.sehan
import ...
Sehan Arandara
class View_Quiz_Details : AppCompatActivity() {
private lateinit var quizId : TextView
private lateinit var quizName : TextView
private lateinit var quizCourse : TextView
private lateinit var quizGrade : TextView
private lateinit var btnUpdate : Button
private lateinit var btnDelete : Button
Sehan Arandara
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContentView(R.layout.activity_view_quiz_details)
quizId = findViewById(R.id.viewQuizId)
quizName = findViewById(R.id.viewQuizName)
quizGrade = findViewById(R.id.viewGrade)
quizCourse = findViewById(R.id.viewCourse)
btnDelete = findViewById(R.id.deleteQuizBtn)
btnUpdate = findViewById(R.id.updateQuizBtn)
initView()
setValuesToViews()
Quiz Informations
Quiz id
-NUgb5cENl1oS7kIBo61
Quiz Name
Quiz 01
Course
10
Grade
c
EDIT QUIZ
DELETE QUIZ



3. Update Operation



4. Delete Operation



3. IT21166174 (Navindi R.L.S)- Book Store management

I am handling the Books management.

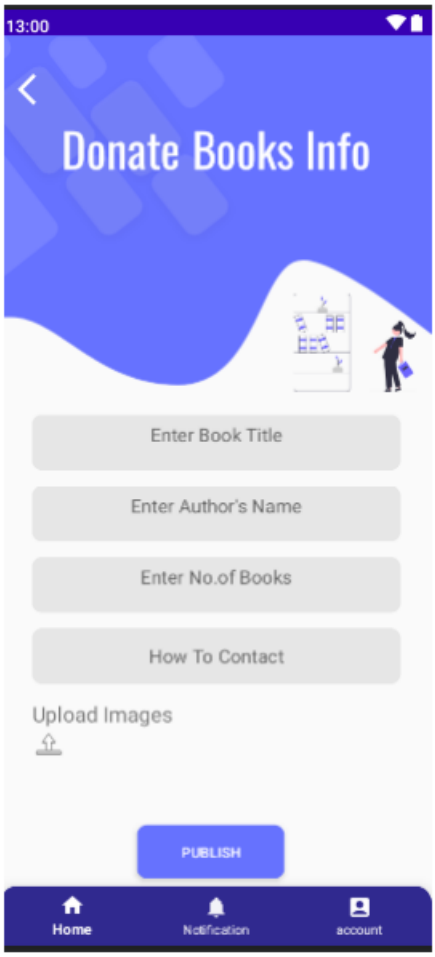
Principles

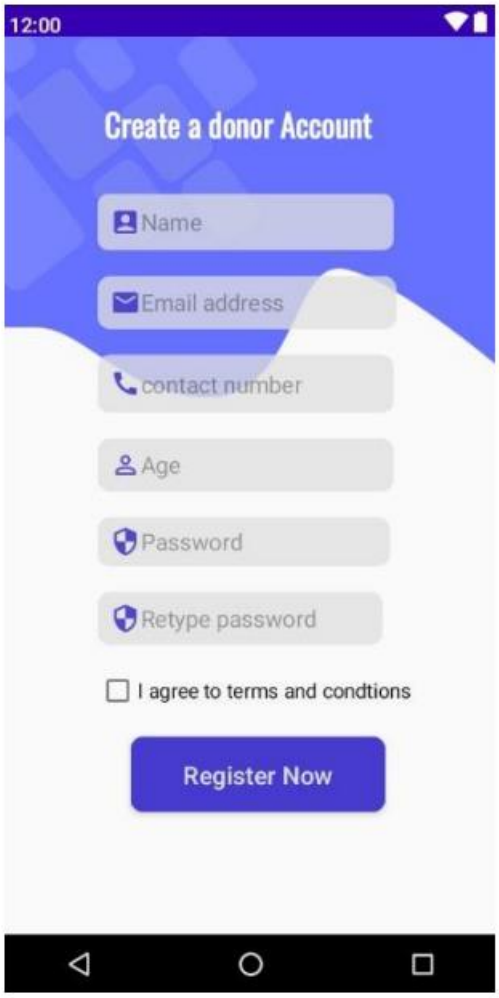
I have used be focused, be charming mobile mindset principles, and I used global guidelines principles such as responsiveness, and polish. Also, to get a good first impression I have used different buttons, images.

CRUD OPERATIONS

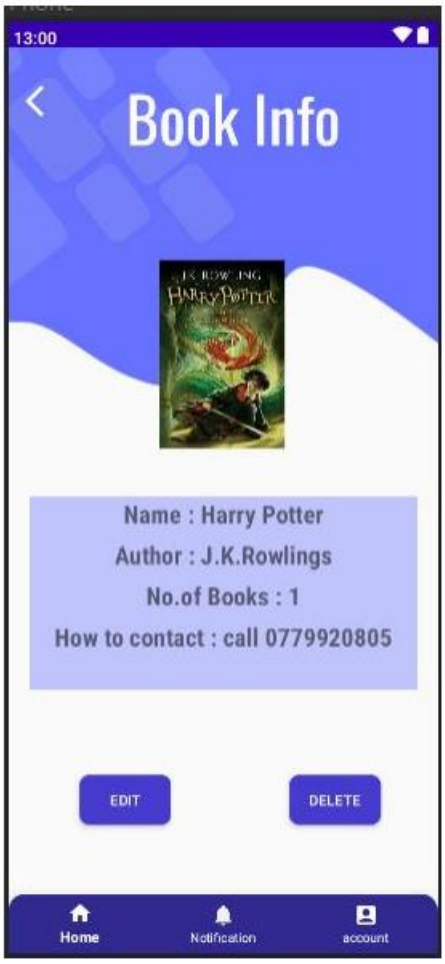
This page includes all the donated book information. The donated books are inserted and uploaded via this page. Users can view the book info as they required. Admin can update or delete any book using “Edit” and “Delete” buttons respectively if necessary. Only admin can edit and delete the details about books in this system. A user can retrieve the books that they have already published after publishing a new book using “Publish” button. And also donators can add books to the system and their contact details to donate their books physically. There is a calculation function used to show the book count that has been donated by a donator. All the data inserted are stored in Firebase database.

1. Create Operation





2. Read Operation



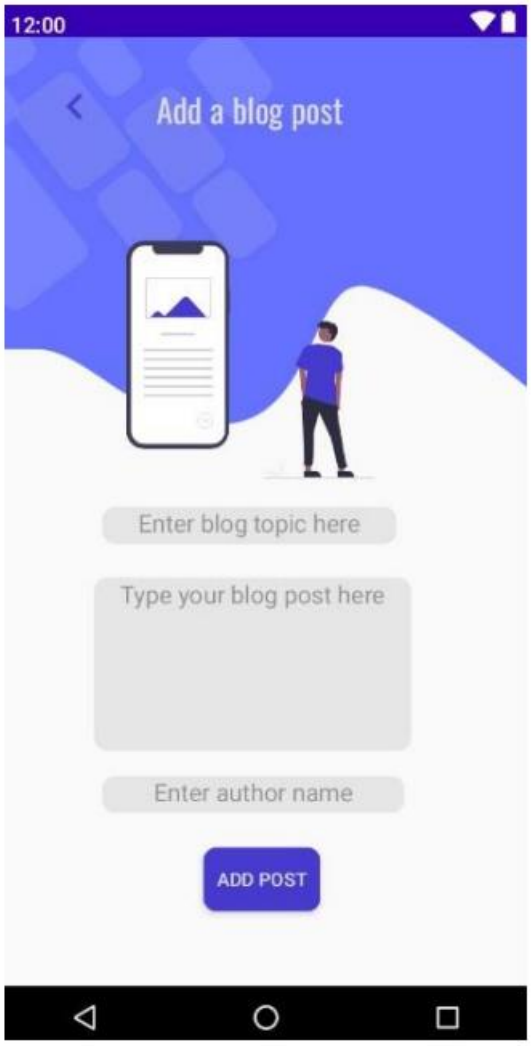
4. IT21184444 (Sumanasekara W.H.U.)– Blog management

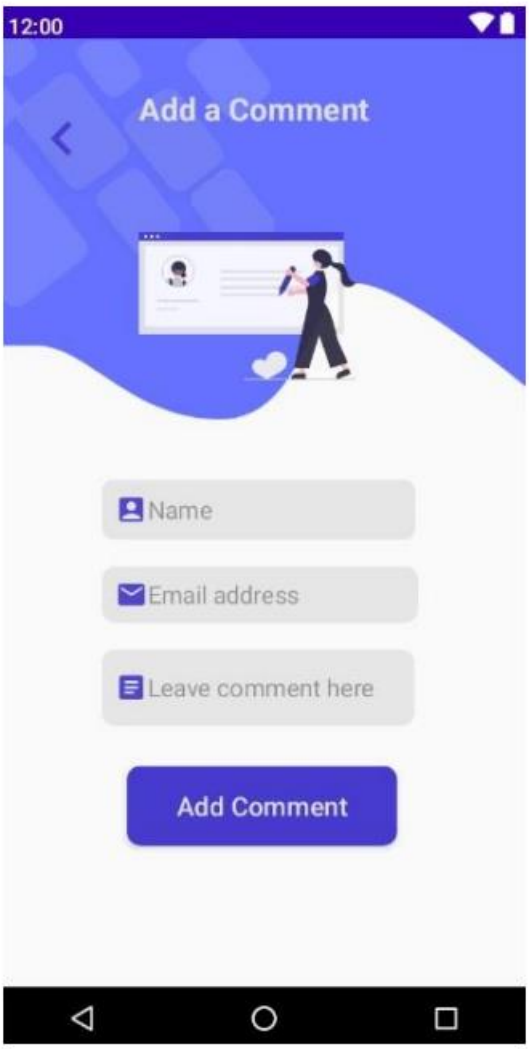
In our EduX app I am the responsible for the blog management feature, which allows users to create, read, update, and delete (CRUD) blog posts. Users can create blog posts by filling in the topic and blog area. Once they submit their post, it will be saved to the app's database, and other users can see it. Other users can search for and read blog posts created by other users. They can also view their own blog posts and all the comments made about their own blog entries.
To provide more flexibility to the users, the app allows them to edit and delete their blog posts. They can edit the topic and blog area of their post and save the updated version. When a user no longer wants a particular blog post to be displayed on the app, they can delete it. The user can choose the post they wish to permanently remove from the database and then confirm their deletion choice. Additionally, users can add comments to other blog posts and edit their own comments as needed. This feature encourages interaction and feedback among users. To ensure the security and integrity of the app, appropriate permissions and access control will be

implemented. This will enable users to only edit and delete their own content and limit access to sensitive data to authorized personnel only.
Overall, these CRUD procedures give users the freedom to maintain their blog entries and interact with other users' content, encouraging interaction and sharing of knowledge within the app.

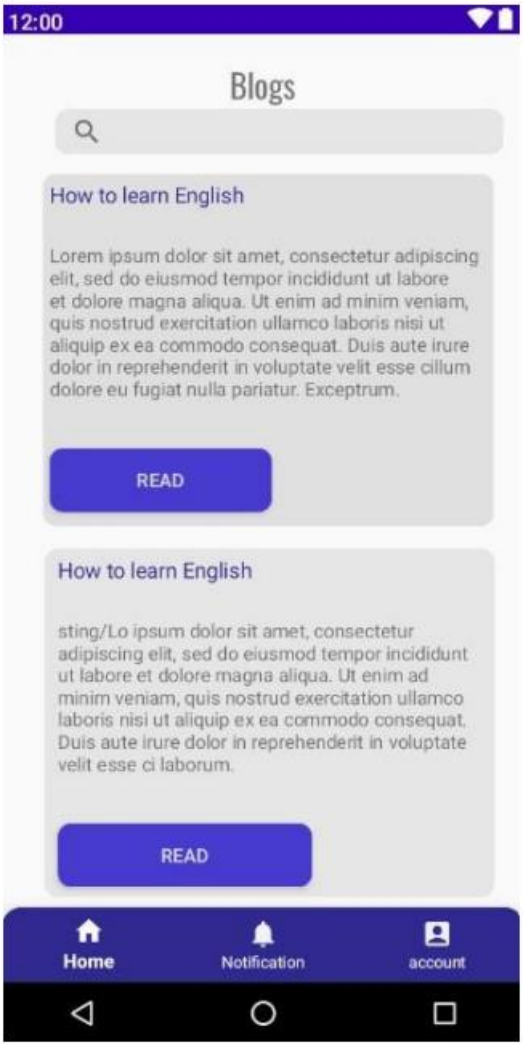
CRUD OPERATIONS

1. Create Operation




12:00
Add a Comment
Name
Email address
Leave comment here
Add Comment

2. Read Operation




12:00
Blogs
How to learn English
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Exceptrum.
Leave a Comment