Android Project (20%):

Student Course Booking App

SEG2105 – Introduction to Software Engineering

Fall 2022

University of Ottawa

uOttawa

Course Coordinator: Dr. Omar Badreddin

Group 19

Rahul Gour – 300271724

Burak Toprak – 8911350

Koustubh Rachuri – 300244225

David Nguyen – 300232884

Due Date: 2022-12-05

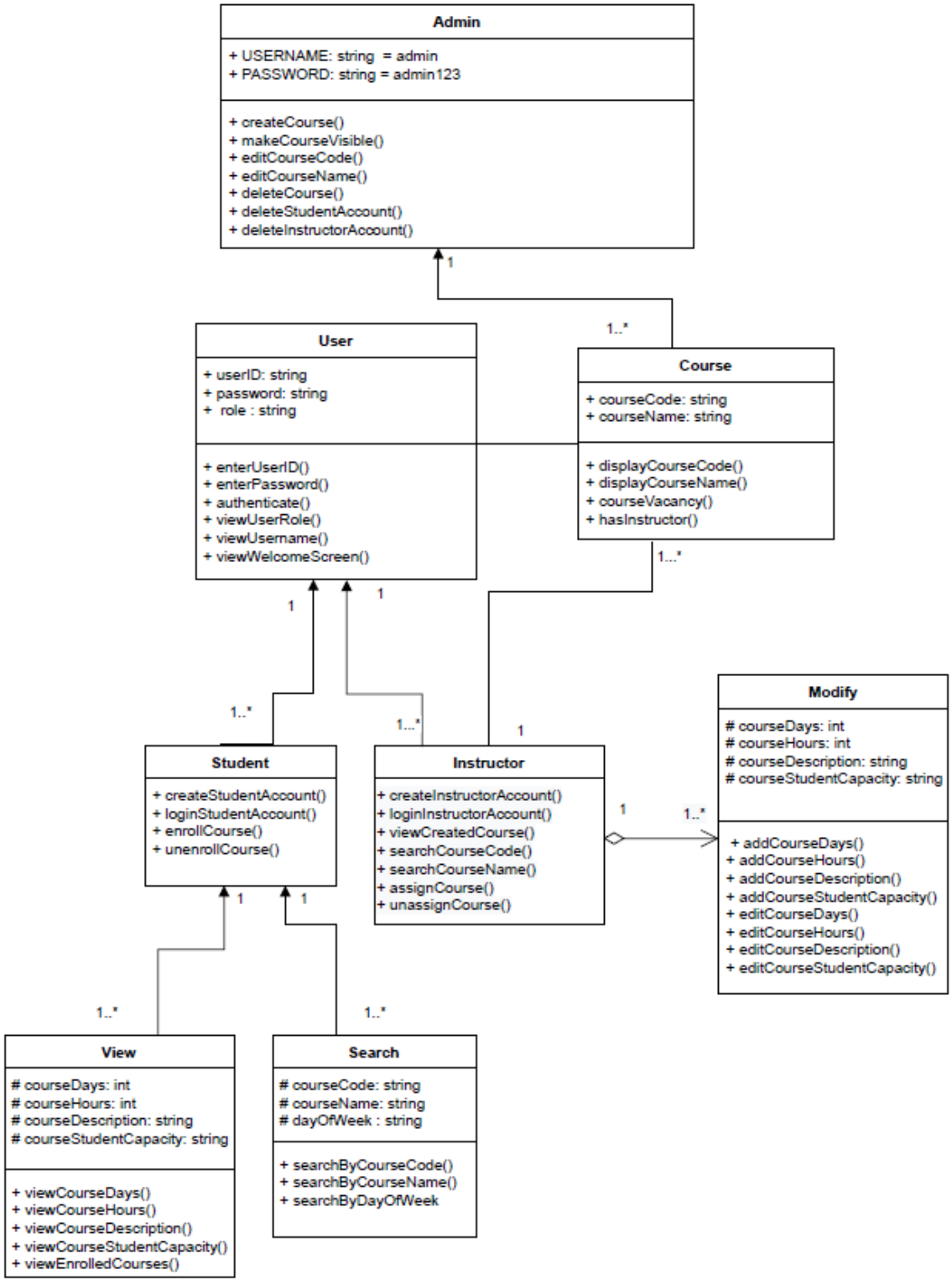
Introduction

This report shows the progress of the final "Student Course Booking App" project for this course. Students will be able to search for a course, instructors can edit details about their own course, and an administrator will manage users and courses.

The creation of this project is done through 3 deliverables:

1. Deliverable 1: Implemented the user account management component and the admin functionalities. Constructed UML Diagram to show the needed class, methods, and instance variables and the different events that occur throughout a user utilizing the application.
2. Deliverable 2: Implemented the instructor functionalities. Updated UML Diagram with Instructor Settings.
3. Deliverable 3: Implemented the student functionalities and finalized the entire application. Finalized UML Diagram with all functionalities.

Final UML Class Diagram


Admin
+ USERNAME: string = admin
+ PASSWORD: string = admin123
+ createCourse()
+ makeCourseVisible()
+ editCourseCode()
+ editCourseName()
+ deleteCourse()
+ deleteStudentAccount()
+ deleteInstructorAccount()
1
1..*
User
+ userID: string
+ password: string
+ role : string
+ enterUserID()
+ enterPassword()
+ authenticate()
+ viewUserRole()
+ viewUsername()
+ viewWelcomeScreen()
Course
+ courseCode: string
+ courseName: string
+ displayCourseCode()
+ displayCourseName()
+ courseVacancy()
+ hasInstructor()
1...*
1
1
1..*
1...*
1
Modify
courseDays: int
courseHours: int
courseDescription: string
courseStudentCapacity: string
+ addCourseDays()
+ addCourseHours()
+ addCourseDescription()
+ addCourseStudentCapacity()
+ editCourseDays()
+ editCourseHours()
+ editCourseDescription()
+ editCourseStudentCapacity()
Student
+ createStudentAccount()
+ loginStudentAccount()
+ enrollCourse()
+ unenrollCourse()
Instructor
+ createInstructorAccount()
+ loginInstructorAccount()
+ viewCreatedCourse()
+ searchCourseCode()
+ searchCourseName()
+ assignCourse()
+ unassignCourse()
1
1..*
1
1
1..*
1..*
View
courseDays: int
courseHours: int
courseDescription: string
courseStudentCapacity: string
+ viewCourseDays()
+ viewCourseHours()
+ viewCourseDescription()
+ viewCourseStudentCapacity()
+ viewEnrolledCourses()
Search
courseCode: string
courseName: string
dayOfWeek : string
+ searchByCourseCode()
+ searchByCourseName()
+ searchByDayOfWeek

Contributions

Members	Deliverable 1	Deliverable 2	Deliverable 3
Rahul Gour	%25	%25	%25
Burak Toprak	%25	%25	%25
Koustubh Rachuri	%25	%25	%25
David Nguyen	%25	%25	%25

Rahul Gour elaborated with MyDBHandler, RegisterPage, StudentPanel java classes.

Burak Toprak elaborated with xml and UI of the application.

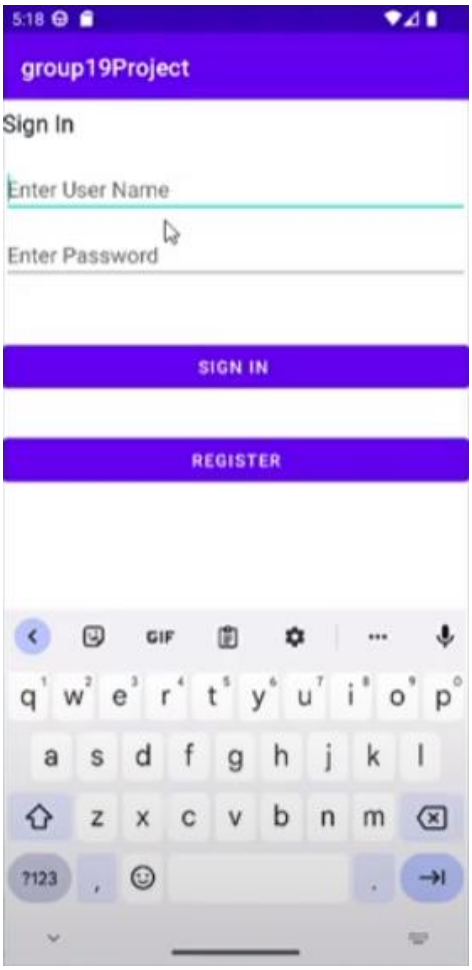
Koustubh Rachuri elaborated with ActivityDashboard, Adminpanel, Course and Information java classes.

Davidy Nyugen elaborated with InstructorPanel, MainActivity and Validity java classes.

Screenshots of UI

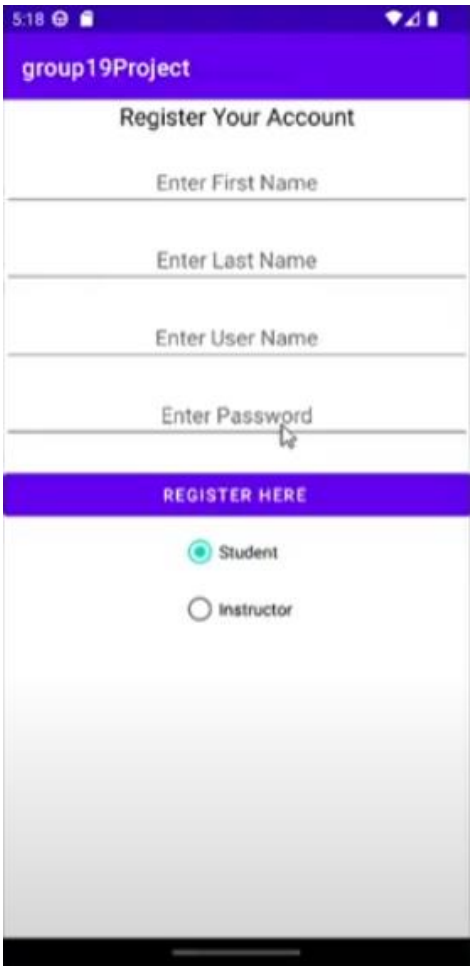
Sign In

The user can sign in with their existing Account. This menu is created by using MainActivity.java class.



Register

The user can register as an instructor or as a student. This menu is created by using RegisterPage.java class.



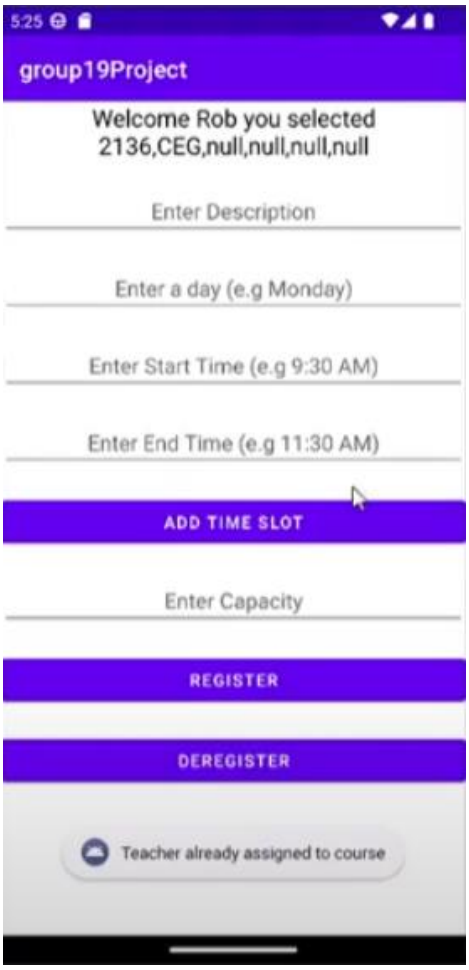
Admin Menu

Admin can add, update, delete courses and delete users on this page. You can access it with only admin login credentials. This menu is created by using AdminPanel.java also MyDBHandler.java to get course and user information. In this menu, we are proud of a feature that we implemented in the app;

When you add a new course, it displays under the function buttons. This way it is easier for the user to navigate within the menu. You can see it in the second screenshot.

Instructor Menu

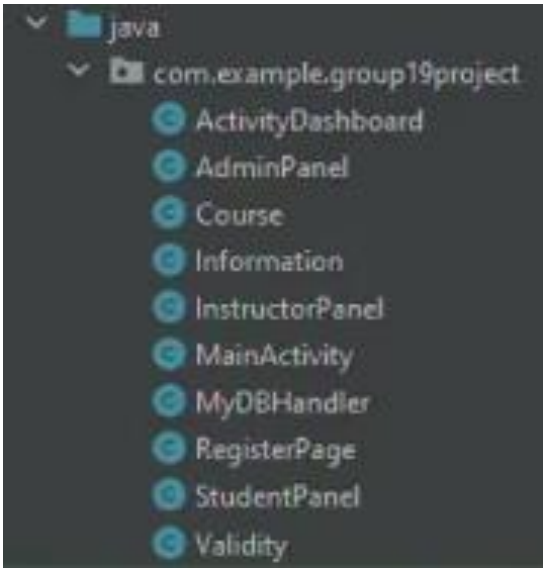
Instructors can add new courses, update, and delete courses on this page. In the second screenshot, Instructors can add description, Start date and End Date for the course. This menu is created by using InstructorPanel.Java class and MyDBHandler.java class.



Student Menu

Students can view all the available courses and enroll in the courses. If they enroll in any course, they can also unenroll from the menu. This menu is created by using StudentPanel.java class.

All Java Classes



Activity Dashboard.java

After the user sign in. It directs to a new layout and displays the user role (student, instructor, or admin). It gets user and account type information from MainActivivity.java

AdminPanel.java

In this class we have functions like add, update, edit, or delete. We store and retrieve information from database MyDBHandler.java class.

Course.java

Provides Course Information, courseName, courseID, instructor, description and capacity.

Information.java

Provides User Information, firstName, lastName, userName, password, accountType, courses and timeBusy.

Instructor Panel.java

In this class we have functions like add, update, and delete. We can add or delete new courses and update course information. We send information to database and pull information from database MyDBHandler.java class.

MainActivity.java

We have Sign-in and Register Functions. Also check database if the user exists and write new user registration to database MyDBHandler.java.

MyDBHandler.java

This is our database where we save and display the input from the user for Courses and User information. We used 2 tables, the first table is for User Accounts and the second table is for Course Information.

RegisterPage.java

If the user does not have an existing account, it gives an error, and the user can create a new account with their assigned role (student, instructor)

StudentPanel.java

It displays available courses for the student and the student can enroll or unenroll from the course. It retrieves data from MyDBHandler.java.

Validity.java

It validates input information from the user if the input is valid or not, validDay,validTime,validCapacity.

Database

We were provided by 2 databases from Dr. Omar Badreddin in the assignment, Firebase and SQLite. It was crucial for the assignment to focus on using SQlite database because it is a local database on Android device (data stored/processed on a device) with SQL interface. We used 2 tables, table 1 is for userAccounts and table 2 is for courseTable. In the first table we used it for User Information, userAccounts,firstName,lastName,userName,password,accountType. In the second table we used it for Course Information, courseCode, courseName,courseInstructor,courseDescription, courseCapacity and courseTime.

Challenges

Making an effort a group and aiming for the best for each other is one important task we focused on to become a team while doing our parts in this assignment. Each individual faced many challenges, but most challenges were solved with the team effort! One challenge we all had to take a close look at was to implement multiple table on the database. Figuring out the implementation took a lot of studying and learning. Perhaps, it was one of the parts which took most of our time with the project. Accomplishing this challenge was an experience for all of us and it surely stood out in our project.