CSCI 524: Mobile Application Design/Dev

Assignment-3 Course Registration App (Option A)

Name: Rahul Ananda Bijai **STFX ID:** 202006042

Introduction:

Continuing with Assignment 2 on course registration app which could only register for single user, now can be used to register multiple users and these users can register to different courses with all the same conditions as specified on the guidelines requested. User first needs to register by filling the registration form which requires, user's name, id, and password. Once the registration is successful then user can use the user id and the password to login to the app and register for the courses that they want to register. App has the list of courses that a particular user can select and see the details for the course that has been selected. User can register for the courses if he/she meets all the prerequisite needed to register for that course. User can register for maximum of three courses per term. App has 2 courses already registered which are mandatory. User can unregister from courses as well. An alert message will pop up when user tries to unregister any courses until he confirms the action to be performed. User can also see all the courses that they are registered to by clicking on the "Registered Courses" button on the home page. App also has an additional functionality for admins. Admins have their own login page which they can use to login to the app and can add a course, delete a course, or even modify the existing course that the users can register to. Admin can also view all the users that have registered to their respective courses.

IMPORTANT NOTE:

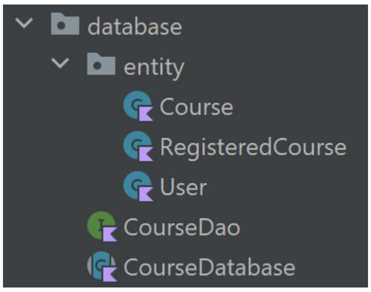
- **Admin Functionality is an additional functionality** which is not required as per the requirements in Assignment 3. It is not fully optimised to work with the user data or user's registered courses.
- If you want to test the admin functionalities with the users, **please make changes with the admin first**, like adding, delete or modifying the course and then shifting to user to see how it reflects the user's homepage.
- **Admin Id and Password for Admin is** "admin"

App's Design & Implementation:

This app consists of following components:

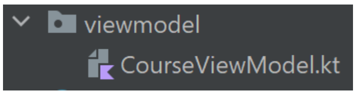
- **3 Entity Data Classes:**
  - o **Course (course_table):** It has 'courseId', 'courseName', 'courseTerm', 'coursePrerequisite' and 'courseType' columns with 'courseId' and 'courseTerm' together defined as a composite key for the table.

  - **RegisteredCourse (registered_course_table):** It has ‘id’ as primary key with ‘userId’, ‘courseId’, ‘courseName’, ‘courseTerm’, ‘coursePrerequisite’ and ‘courseStatus’ columns.
  - **User (user_table):** It has ‘userId’, ‘userName’, ‘userPassword’ and ‘userRole’ columns with ‘userId’ as primary key.
- **1 Data Access Object:**
  - **CourseDao:** This provides access to the data in the SQL tables. This contains 9 queries, 3 insert methods, 2 delete methods and 1 update methods to work with the data in the tables.
- **1 Database**
  - **CourseDatabase:** This class specifies which entities are defined in the database and provides access to a single instance of each DAO class. It also pre-populates the database from assets directory.



```
database
  entity
    Course
    RegisteredCourse
    User
  CourseDao
  CourseDatabase
```

- **1 Prepacked Database in Assets Directory:**
  - **course_database.db:** This .db file is created with the help of the software called “DB Browser for SQLite” and I had the courses details inserted in this database which is loaded when the apps is initialized.
- **1 View Model:**
  - **CourseViewModel:** In this view model, multiple functions and methods are created to call the functions from CourseDao and process it and passed to different fragments that call these functions from view model. This helps in keeping the UI data persisted.

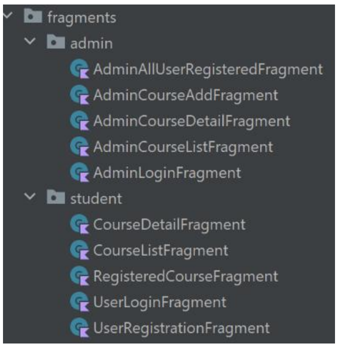


```
viewmodel
  CourseViewModel.kt
```

- **10 Fragments:**
  - **UserRegistrationFragment**: This fragment is used for adding new users to the user_table in the database so that the user can then login to app for course registration.
  - **UserLoginFragment**: This fragment validates the user id and password of Users from the user_table and the registered users can enter their user id and user password to login to the app.
  - **CourseListFragment**: This fragment calls the recycler view and its list adapter for showing the list of courses. It also consists of a button which performs the action to go to “CourseRegisteredFragment” to see the list of courses that have been registered.
  - **CourseDetailFragment:** In this fragment, user can see the course name, id, prerequisite, term, and its status. There are two buttons in this fragment, one

is for registering the course and other is to navigate back to the home page. This fragment also has the function for checking the conditions of registering the course.

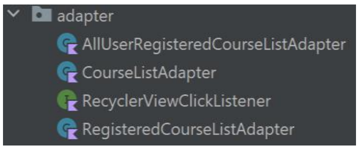
- **RegisteredCourseFragment:** This fragment calls the recycler view and its list adapter for displaying the list of courses that has been registered by the user who have logged in. User can also un-register from the courses in this fragment. It also contains function for checking the conditions for unregistering a particular course.
- **AdminLoginFragment:** This fragment validates the admin id and password of Admins from the user_table and only admins will be able to login using their id and password.
- **AdminCourseListFragment:** This fragment calls the recycler view and its list adapter for showing the list of courses. It also consists of a button which performs the action to go to "AdminAllUserRegisteredFragment" to see the list of courses that have been registered by all users using this app. There is a floating button that navigates to "AdminCourseAddFragment" for the admin to add new courses.
- **AdminCourseDetailFragment:** In this fragment, admins can see the course name, id, prerequisite, term, and its type. There are two buttons in this fragment, one is deleting the course from the course_table and other is a floating button to edit the current course that has been opened.
- **AdminCourseAddFragment:** In this fragment, admins can either add a new course or edit the existing course and save it using the save button which performs the write action to the course_table.
- **AdminAllUserRegisteredFragment:** In this fragment, admins can view all the users details who have registered to any of the courses. These details include user id, course name, course id and course term.



```
fragments
  admin
    AdminAllUserRegisteredFragment
    AdminCourseAddFragment
    AdminCourseDetailFragment
    AdminCourseListFragment
    AdminLoginFragment
  student
    CourseDetailFragment
    CourseListFragment
    RegisteredCourseFragment
    UserLoginFragment
    UserRegistrationFragment
```

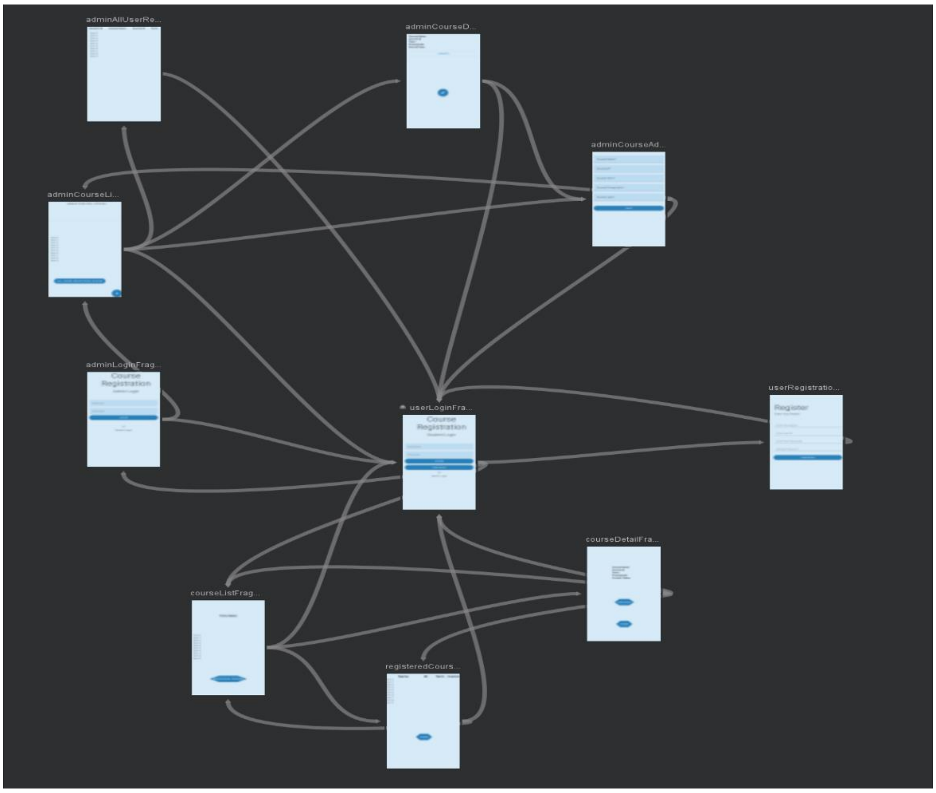
- **<u>3 List Adapter for 3 Different Recycler Views:</u>**
  - **CourseListAdapter:** Used by the recycler view that is called in "CourseListFragment". It gets the data of the list of courses from the view model as a live data and displays them. On clicking the item on this recycler view, an action has been performed to navigate to the "CourseDetailsFragment" along with 3 arguments (Course ID, Term and Student ID) passed with it to the other fragment. "course_view.xml" is created to show how each item in this recycler view should be displayed.

  - **RegisterCourseListAdapter:** Used by the recycler view that is called in "RegisteredCourseFragment". It gets the data of the list of courses that has been registered by the student from the view model as a live data and displays them. There is Image Button for students to click to un-register from the course. "register_view.xml" is created to show how each items in this recycler view should be displayed.
  - **AllUserRegisteredCourseListAdapter:** This adapter is used by admin to display all the courses that has been registered by all users using the app and displaying in the recycler view. "alluser_register_view.xml" is created to show how each item in this recycler view should be displayed.

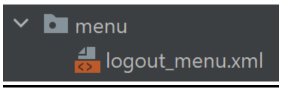


```
adapter
    AllUserRegisteredCourseListAdapter
    CourseListAdapter
    RecyclerViewClickListener
    RegisteredCourseListAdapter
```

- **<u>Navigation:</u>**
  - Used to navigate the pages as per the actions specified in the nav_graph. This is used by "MainActivity" with "NavController" to manage the navigation of the pages.

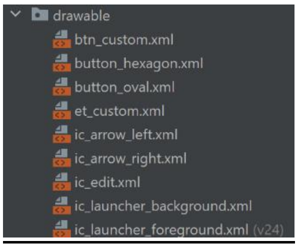


- **<u>1 Menu:</u>**
  - **<u>logout_menu.xml:</u>** Can be used by User or Admin to Logout and return to the login page.



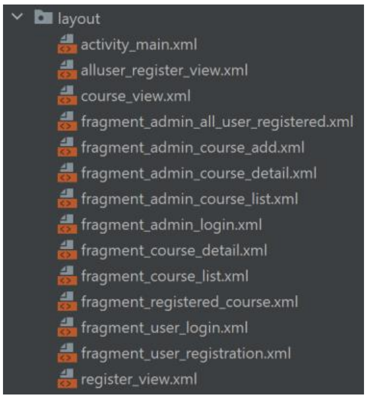
```
menu
    logout_menu.xml
```

- **Drawable:** All the drawable like edit, add, customs designs for buttons and shapes have been created here.



```
drawable
    btn_custom.xml
    button_hexagon.xml
    button_oval.xml
    et_custom.xml
    ic_arrow_left.xml
    ic_arrow_right.xml
    ic_edit.xml
    ic_launcher_background.xml
    ic_launcher_foreground.xml (v24)
```

- **Layouts:**
    - **activity_main.xml:** Contains the Fragment Container View
    - **fragment_user_registration.**xml: UI for user registration fragment. It consists of 2 text view, 4 edit text and 1 button.
    - **fragment_user_login.**xml: Ui for user login fragment. It consists of 3 text view, 2 edit text, 2 buttons and 1 image view.
    - **fragment_course_list.xml:** UI for course list fragment. It consists of 1 recycler view, 1 text view and 1 button
    - **fragment_course_detial.xml:** UI for course details fragment. It consists of 10 text views placed in a linear layout and consist of 2 buttons.
    - **fragment_registered_courses.xml:** UI for registered course fragment. It consists of 1 recycler view, 4 text view and 1 button.
    - **fragment_admin_login.**xml: Ui for admin login fragment. It consists of 3 text view, 2 edit text, 1 button and 1 image view.
    - **fragment_admin_course_list.**xml: UI for admin course list fragment. It consists of 1 recycler view, 1 text view and 2 buttons
    - **fragment_admin_course_detail.**xml: UI for admin course details fragment. It consists of 10 text views and 2 buttons.
    - **fragment_admin_course_add.**xml: UI for admin course add fragment. It consists of 5 text views input layout, 5 edit text and 1 button.
    - **fragment_admin_all_user_registered.xml:** UI for admin all user registered fragment. It consists of 1 recycler view and 4 text view.
    - **course_view.xml:** For recycler view items. It consists of 1 text view.
    - **register_view.xml:** For recycler view items. It consists of 3 text views and 1 image view.
    - **alluser_register_view.xml:** For recycler view items. It consists of 4 text views.

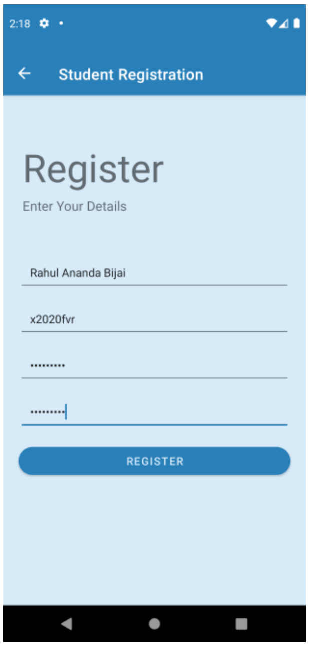


```
layout
    activity_main.xml
    alluser_register_view.xml
    course_view.xml
    fragment_admin_all_user_registered.xml
    fragment_admin_course_add.xml
    fragment_admin_course_detail.xml
    fragment_admin_course_list.xml
    fragment_admin_login.xml
    fragment_course_detail.xml
    fragment_course_list.xml
    fragment_registered_course.xml
    fragment_user_login.xml
    fragment_user_registration.xml
    register_view.xml
```

App’s Functionality & Screenshots:

Student Registration Page:

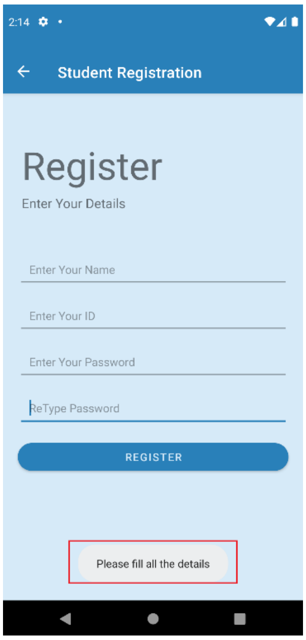
- For Accessing the app, Students need to first register by entering Name, ID, Password and then submitting using the register button.
- If any fields are empty, then a toast message will appear telling the student to fill the all the details
- If the ‘USER ID’ is already registered, then it will show the alert stating that the user already exists.



Rahul Ananda Bijai

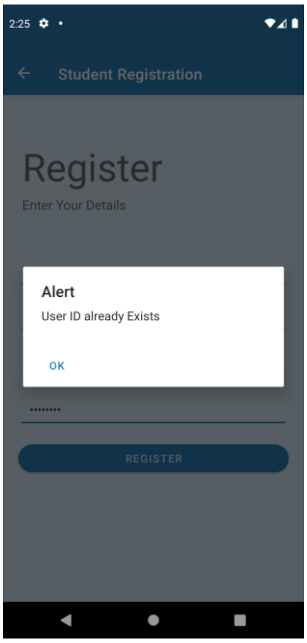
x2020fvr

Successful Register



Please fill all the details

Missed any Field to Enter



Alert

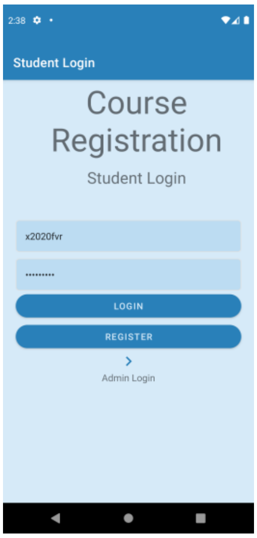
User ID already Exists

OK

User Already Exists

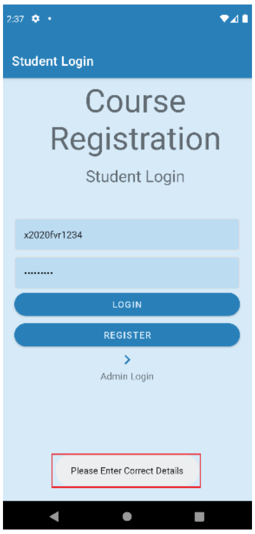
Student Login Page:

- Students can login using student id and password.
- If the details are not correct or if any field is left empty a toast will appear stating to fill the correct details.



x2020fvr

Successful Login



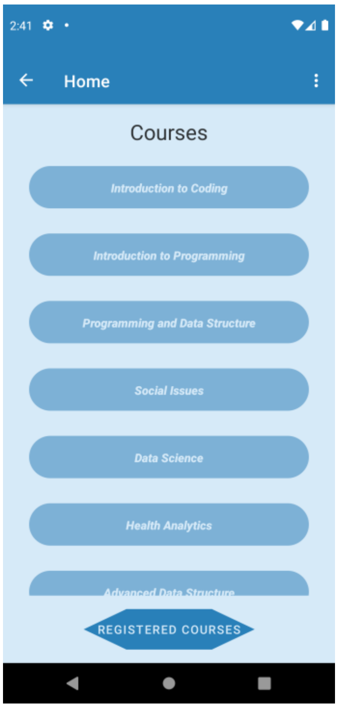
x2020fvr1234

Please Enter Correct Details

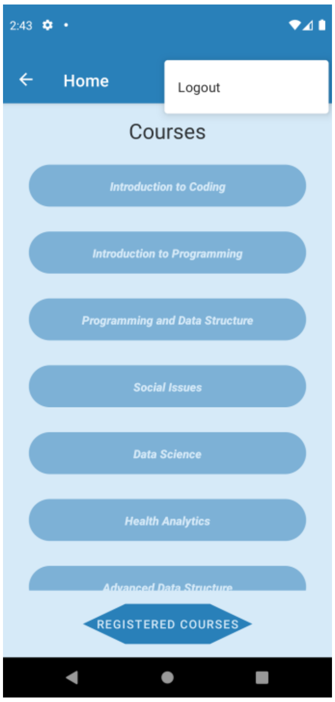
Incorrect/Empty Details

Course List Page:

- User can scroll through the list of courses and select the course they want to see their details for.
- User can also click the “Registered Courses” button which will open the page showing the list of courses they register.
- Students can also logout from the menu on top right.



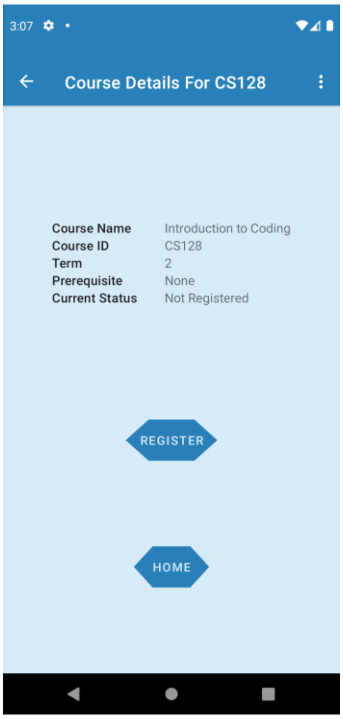
Course List



Logout Option

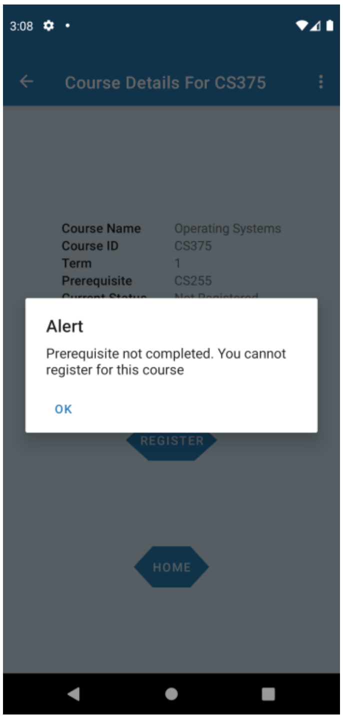
Course Details Page:

- User can see the details of the course that has been selected by them. They have a option to register to the course or go back to home.
- If a student tries to register completed, already registered, term registration full or not satisfactory prerequisite courses then an alert will pop up.
- Students can also logout from the menu on top right.



Course Name Introduction to Coding
Course ID CS128
Term 2
Prerequisite None
Current Status Not Registered

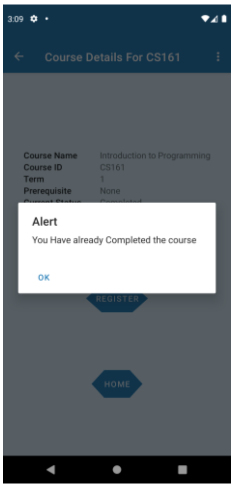
Successful Registration



Course Name Operating Systems
Course ID CS375
Term 1
Prerequisite CS255
Current Status Not Registered

Alert
Prerequisite not completed. You cannot register for this course
OK

Prerequisite Not Satisfied


3:09

Course Details For CS161

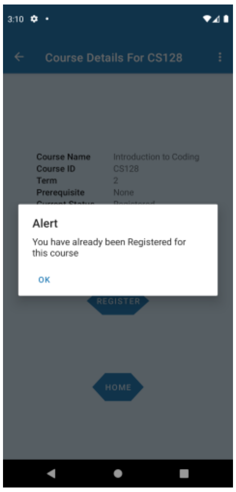
Course Name	Introduction to Programming
Course ID	CS161
Term	1
Prerequisite	None

Alert

You Have already Completed the course

OK

Already Completed


3:10

Course Details For CS128

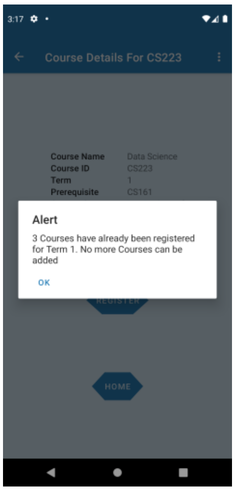
Course Name	Introduction to Coding
Course ID	CS128
Term	2
Prerequisite	None

Alert

You have already been Registered for this course

OK

Already Registered


3:17

Course Details For CS223

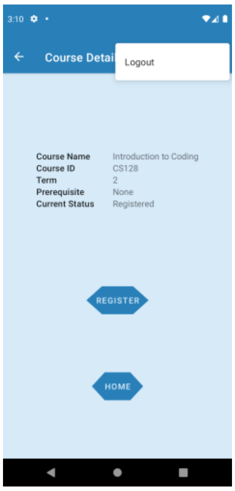
Course Name	Data Science
Course ID	CS223
Term	1
Prerequisite	CS161

Alert

3 Courses have already been registered for Term 1. No more Courses can be added

OK

Term Registration Full


3:10

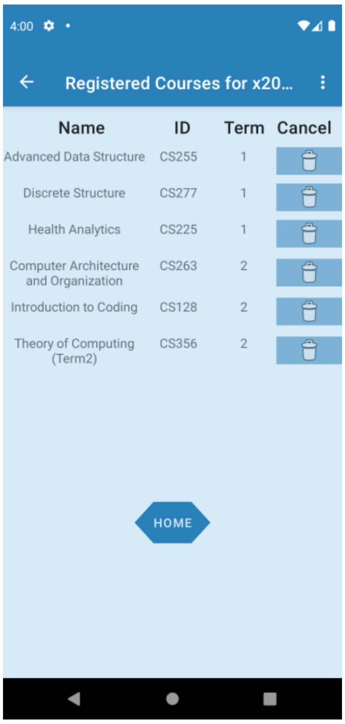
Course Detai

Course Name	Introduction to Coding
Course ID	CS128
Term	2
Prerequisite	None
Current Status	Registered

Logout Option

Registered Course Page:

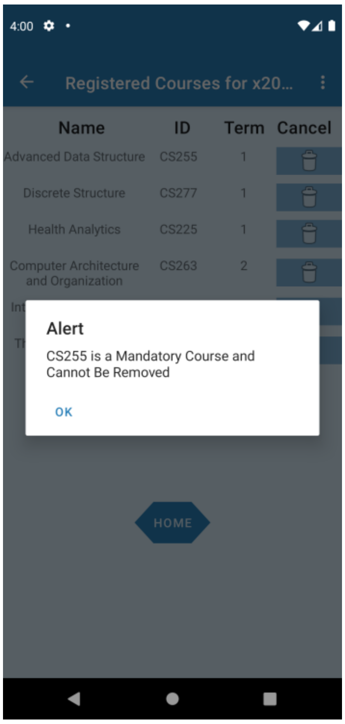
- It displays all the registered courses that a student has registered to.
- They can un-register from any courses which is not mandatory.
- If they un-register a course that is a prerequisite of another course that has been registered, then both courses will be unregistered.
- They will receive an alert box to confirm if they would want to un-register a course.
- Students can also logout from the menu on top right.


4:00

Registered Courses for x20...

Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑
Introduction to Coding	CS128	2	🗑
Theory of Computing (Term2)	CS356	2	🗑

Registered Courses


4:00

Registered Courses for x20...

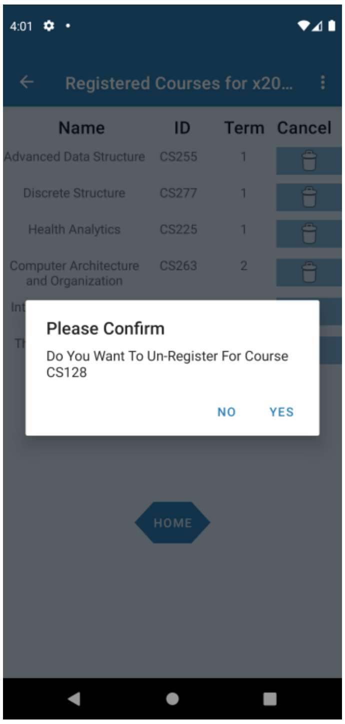
Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑

Alert

CS255 is a Mandatory Course and Cannot Be Removed

OK

Un-Register Mandatory


4:01

Registered Courses for x20...

Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	
Discrete Structure	CS277	1	
Health Analytics	CS225	1	
Computer Architecture and Organization	CS263	2	

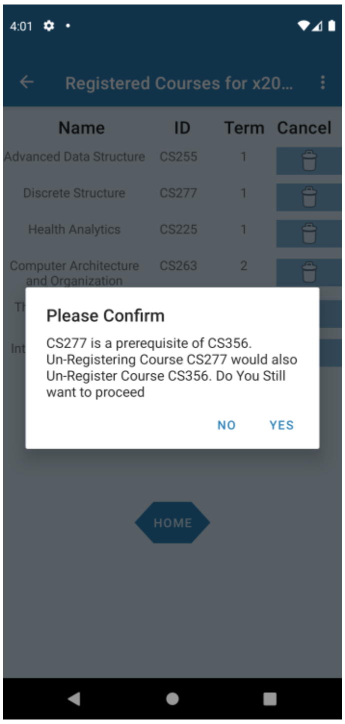
Please Confirm

Do You Want To Un-Register For Course CS128

NO YES

HOME

Un-Register


4:01

Registered Courses for x20...

Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	
Discrete Structure	CS277	1	
Health Analytics	CS225	1	
Computer Architecture and Organization	CS263	2	

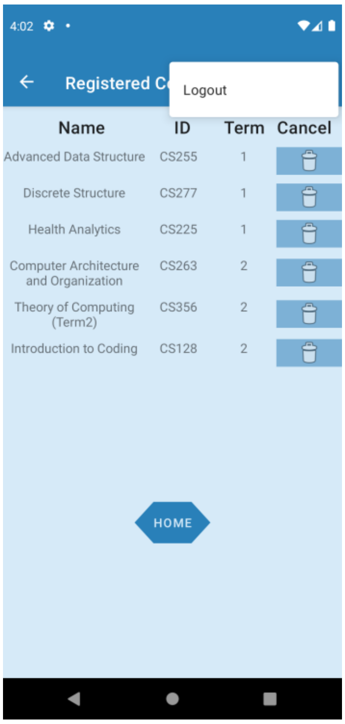
Please Confirm

CS277 is a prerequisite of CS356. Un-Registering Course CS277 would also Un-Register Course CS356. Do You Still want to proceed

NO YES

HOME

Un-Register a prerequisite


4:02

Registered C

Logout

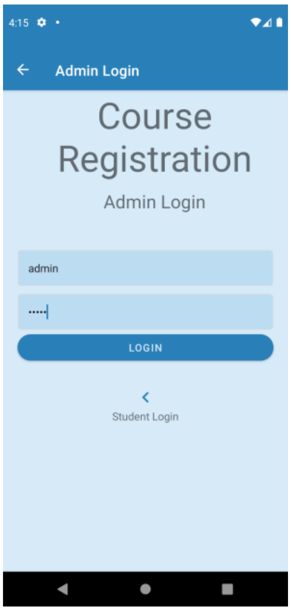
Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	
Discrete Structure	CS277	1	
Health Analytics	CS225	1	
Computer Architecture and Organization	CS263	2	
Theory of Computing (Term2)	CS356	2	
Introduction to Coding	CS128	2	

HOME

Logout Option

Admin (Additional Functionalities):

- Admin can login using its own admin id and password.
- Admins can view the list of courses and see the details of the courses.
- Admins can edit, delete, or add new courses.
- Admins can view all the courses registered by every student who have registered to the app.
- Admins can logout using the menu option on the top right of the page.


4:15

Admin Login

Course Registration

Admin Login

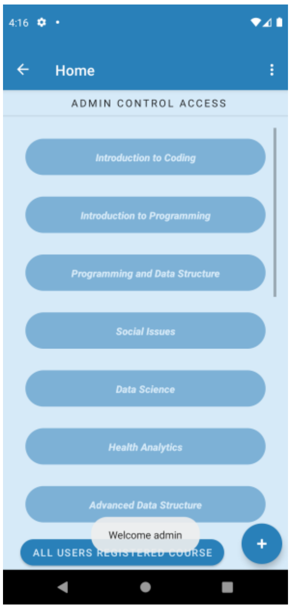
admin

·····

LOGIN

Student Login

Admin Login


4:16

Home

ADMIN CONTROL ACCESS

Introduction to Coding

Introduction to Programming

Programming and Data Structure

Social Issues

Data Science

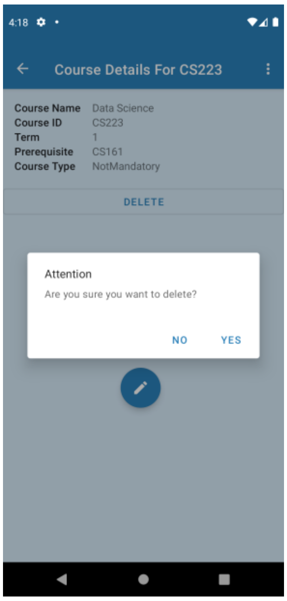
Health Analytics

Advanced Data Structure

Welcome admin

ALL USERS REGISTERED COURSE

Admin Control


4:18

Course Details For CS223

Course Name	Data Science
Course ID	CS223
Term	1
Prerequisite	CS161
Course Type	NotMandatory

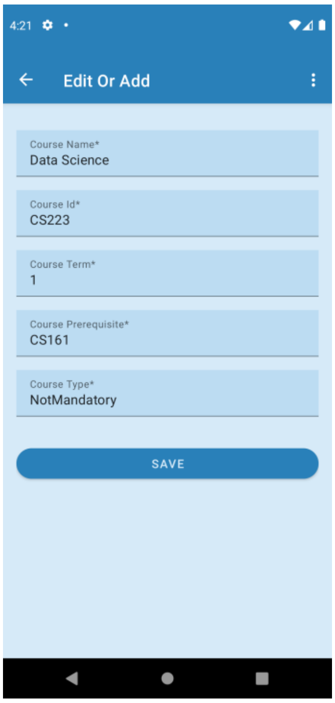
DELETE

Attention

Are you sure you want to delete?

NO YES

Delete Course



Course Name*
Data Science

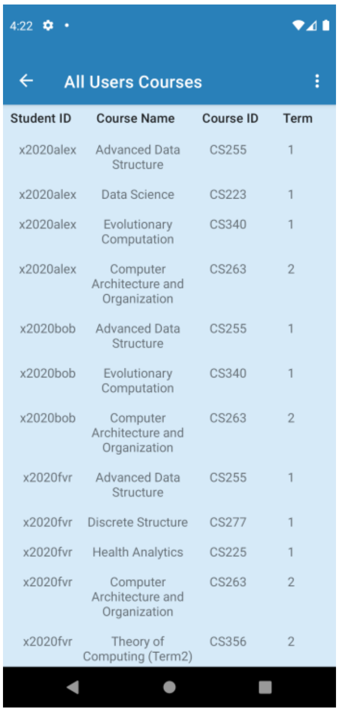
Course Id*
CS223

Course Term*
1

Course Prerequisite*
CS161

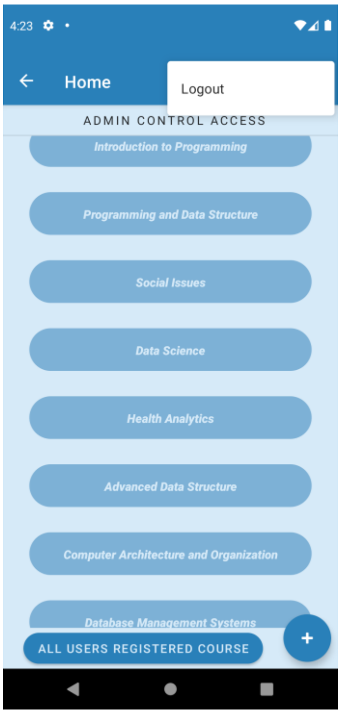
Course Type*
NotMandatory

Add or Edit Course



Student ID	Course Name	Course ID	Term
x2020alex	Advanced Data Structure	CS255	1
x2020alex	Data Science	CS223	1
x2020alex	Evolutionary Computation	CS340	1
x2020alex	Computer Architecture and Organization	CS263	2
x2020bob	Advanced Data Structure	CS255	1
x2020bob	Evolutionary Computation	CS340	1
x2020bob	Computer Architecture and Organization	CS263	2
x2020fvr	Advanced Data Structure	CS255	1
x2020fvr	Discrete Structure	CS277	1
x2020fvr	Health Analytics	CS225	1
x2020fvr	Computer Architecture and Organization	CS263	2
x2020fvr	Theory of Computing (Term2)	CS356	2

View All Students Registered to Courses


4:23
Home
Logout
ADMIN CONTROL ACCESS
Introduction to Programming
Programming and Data Structure
Social Issues
Data Science
Health Analytics
Advanced Data Structure
Computer Architecture and Organization
Database Management Systems
ALL USERS REGISTERED COURSE
+

Logout Option

Others:

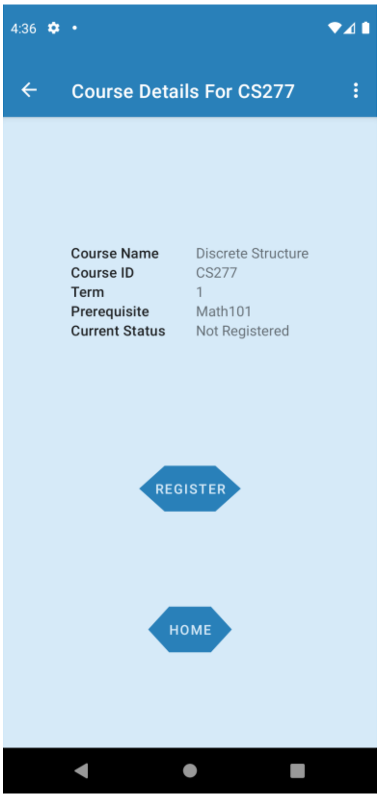
- The UI data is persisted even after the change in state of the activity.
- All the strings have been added to strings.xml.
- All the colour used have been added in colors.xml.
- Primary colour has been changed to blue and light blue to have a pleasant and clear display with text having on black colour.
- Properly labelled each fragment to display the correct name on the page header.

CODE:

Attached the zip file of the app “CourseRegistrationApp.zip” along with this report.

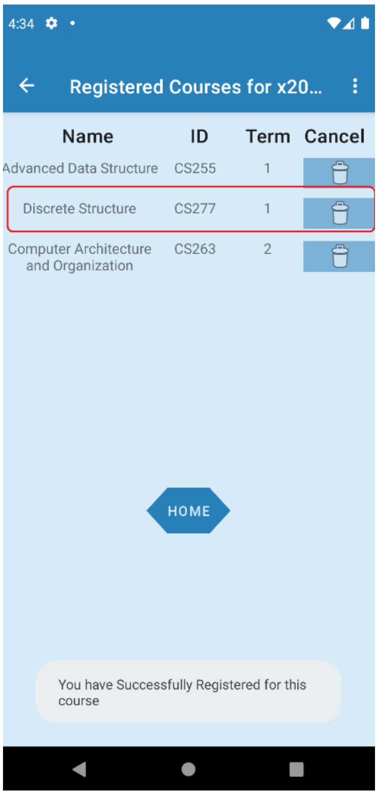
Test Cases & Screenshots Required for Assignment 3:

1.Register for CS277 (Term 1): status: successful



Course Details For CS277

Course Name	Discrete Structure
Course ID	CS277
Term	1
Prerequisite	Math101
Current Status	Not Registered

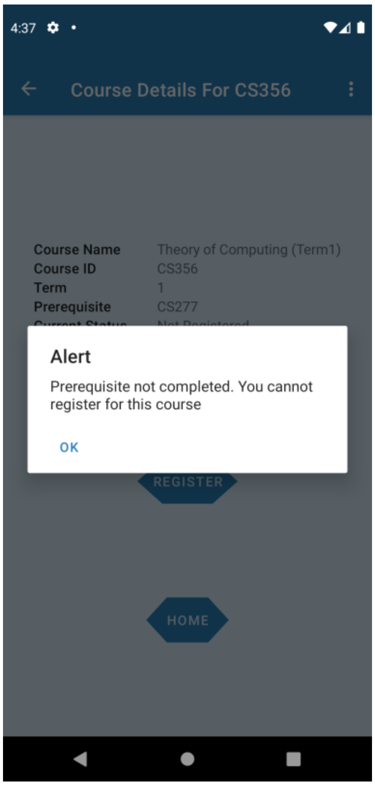


Registered Courses for x20...

Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Computer Architecture and Organization	CS263	2	🗑

You have Successfully Registered for this course

2.Register for CS356 (Term 1): status: unsuccessful due to Prereq

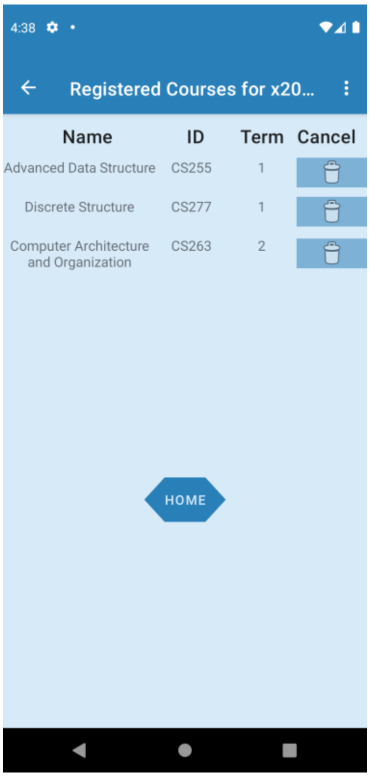


Course Details For CS356

Course Name	Theory of Computing (Term1)
Course ID	CS356
Term	1
Prerequisite	CS277
Current Status	Not Registered

Alert

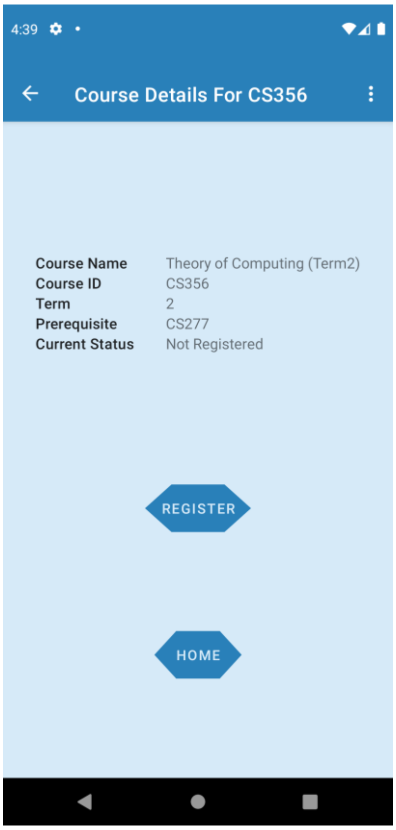
Prerequisite not completed. You cannot register for this course



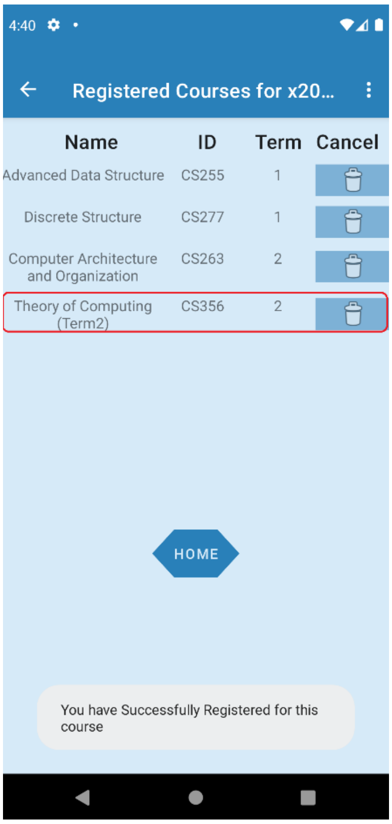
Registered Courses for x20...

Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Computer Architecture and Organization	CS263	2	🗑

3. Register for CS356 (Term 2): status: successful



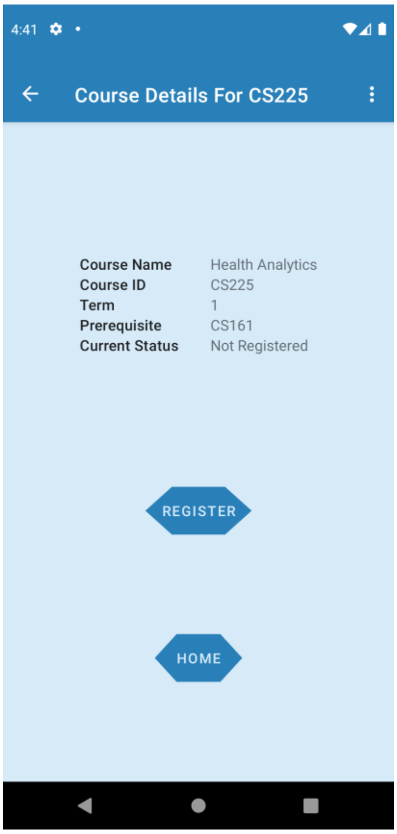
Course Name	Theory of Computing (Term2)
Course ID	CS356
Term	2
Prerequisite	CS277
Current Status	Not Registered



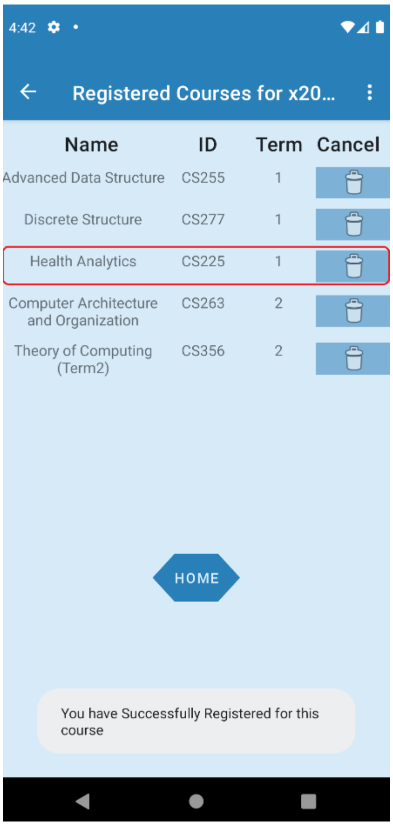
Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Computer Architecture and Organization	CS263	2	🗑
Theory of Computing (Term2)	CS356	2	🗑

You have Successfully Registered for this course

4. Register for CS225 (Term 1): status: successful



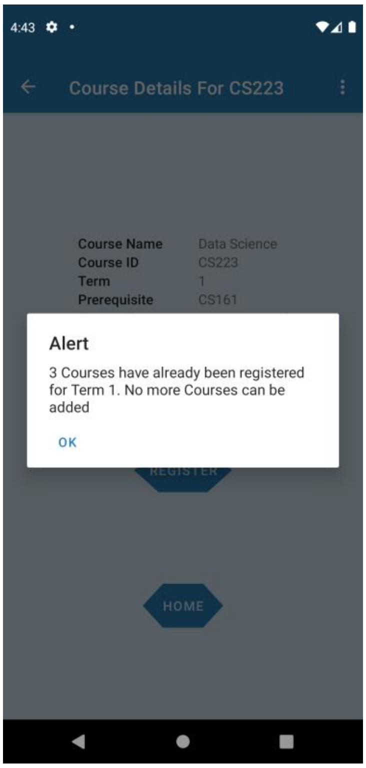
Course Name	Health Analytics
Course ID	CS225
Term	1
Prerequisite	CS161
Current Status	Not Registered



Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑
Theory of Computing (Term2)	CS356	2	🗑

You have Successfully Registered for this course

5. Register another course for Term1: status: unsuccessful due to Limit

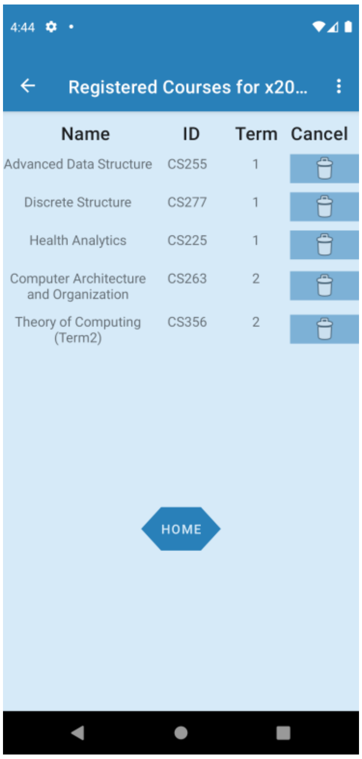


Course Name	Data Science
Course ID	CS223
Term	1
Prerequisite	CS161

Alert

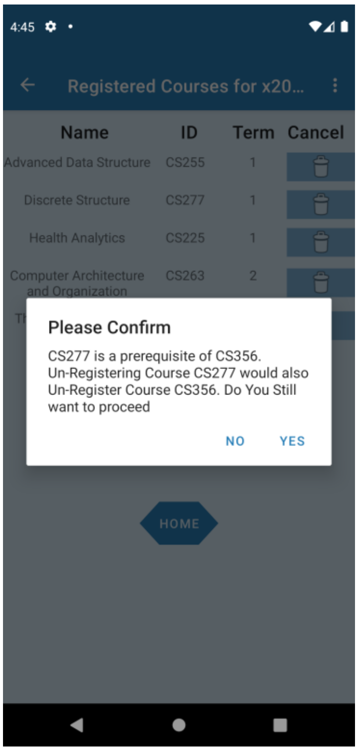
3 Courses have already been registered for Term 1. No more Courses can be added

OK



Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑
Theory of Computing (Term2)	CS356	2	🗑

6. Remove CS277 (Term 1): status: will also remove CS356 (Term 2), Confirm

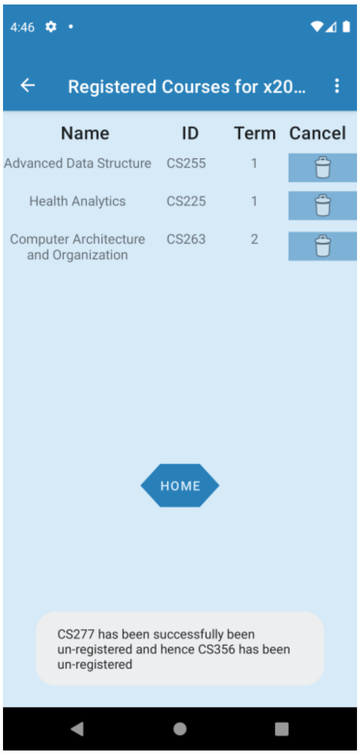


Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Discrete Structure	CS277	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑

Please Confirm

CS277 is a prerequisite of CS356. Un-Registering Course CS277 would also Un-Register Course CS356. Do You Still want to proceed

NO YES



Name	ID	Term	Cancel
Advanced Data Structure	CS255	1	🗑
Health Analytics	CS225	1	🗑
Computer Architecture and Organization	CS263	2	🗑

CS277 has been successfully been un-registered and hence CS356 has been un-registered