



**UNIVERSITI MALAYSIA TERENGGANU**

**UNIVERSITY MALAYSIA TERENGGANU**

**FACULTY OF OCEAN ENGINEERING TECHNOLOGY & INFORMATICS**

**[CSM3114]**

**FRAMEWORK BASED MOBILE APPLICATION DEVELOPMENT**

**(GROUP 1)**

**FINAL REPORT PROJECT 2:**

**E-COURSE APP**

**PREPARED BY:**

**Rabiatul Binti Sulaiman (S62651)**

**PREPARED FOR:**

**DR MOHAMAD NOR BIN HASSAN**

***[BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONOURS]***

**SEMESTER I 2023/2024**

## Contents

Executive summary .....	3
Use Case.....	4
The common structure of tree widgets.....	5
Flutter widget and features adopted in the application .....	6
Sample of Interface with the explanation .....	8
Conclusion .....	11
Reference .....	12
GitHub.....	13

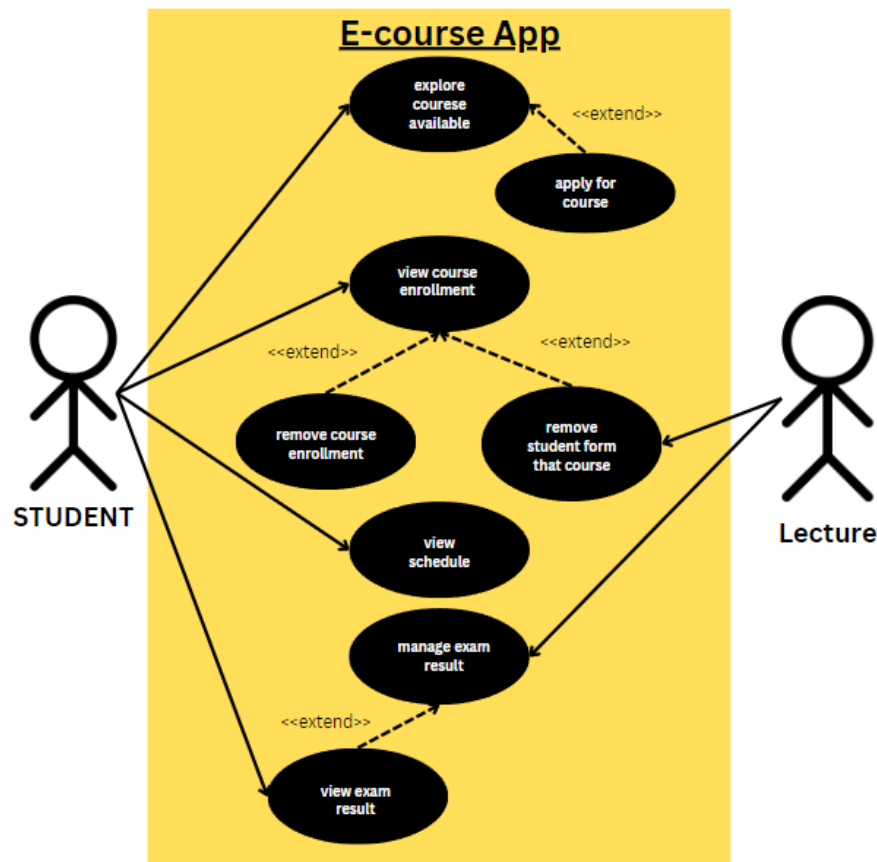
## **Executive summary**

This e-course application was created to make it easier for students to register for courses online without location and time limitations. It provides facilities such as login, add courses, schedules, registration, and exam results. Students can manage all aspects of their course without difficulty, from registration to monitoring exam results, all in this e-course application.

Lecturers can monitor registered students and manage technical problems by removing students from courses manually. This application also coordinates lecturers in easily updating exam results scores.

With reference to Universiti Malaysia Terengganu's MyNemo and the MC Plus application, I have chosen a simple and easy-to-understand interface design, suitable for students and lecturers of various ages. The use of a database also ensures easy accessibility of information and ease of updating. Overall, this e-course application aims to provide a smooth and efficient experience for students and lecturers in their course management.

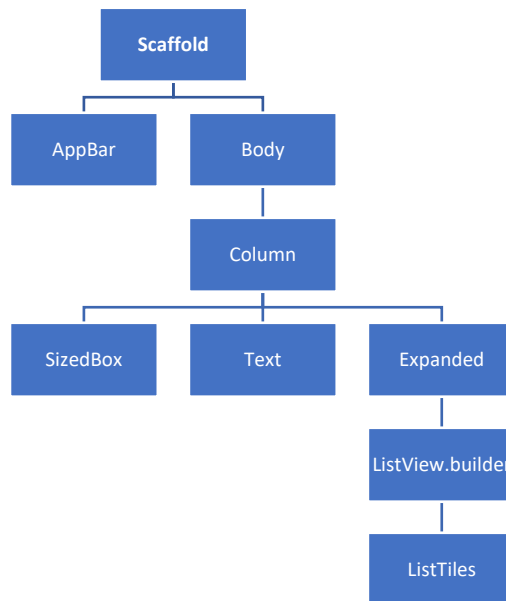
## Use Case



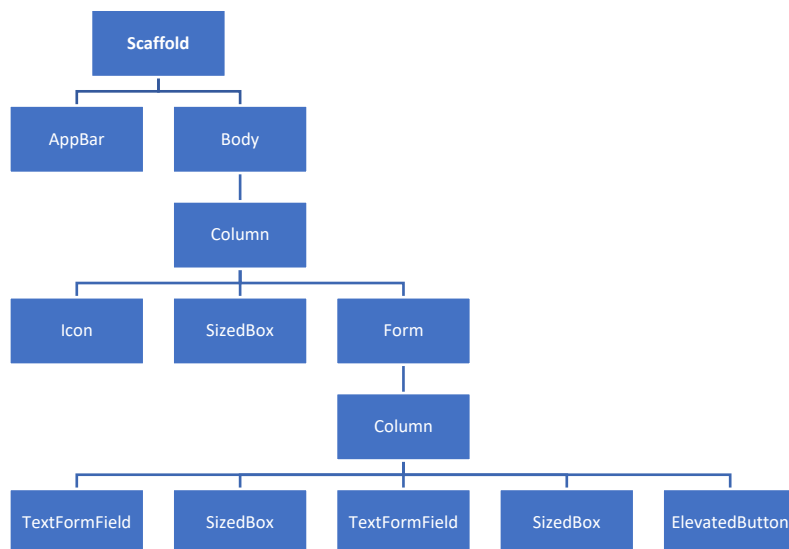
This e-course application offers students the facility to browse available courses through a user-friendly interface or search for specific courses based on their preferences. When finding a course of interest, students can click on it, provide the required information, and proceed to register for the course. Additionally, students have the option to view a list of their registered courses and can delete any registration if necessary. Once all courses are registered, students can access their schedule to see the timetable of the registered courses. Next, students can check whether the course results have been sent by their respective lecturers.

For lecturers, the platform provides tools to manage registered students, giving them the power to remove students from courses they teach. Lecturers are also responsible for updating and maintaining exam results on the platform.

## The common structure of tree widgets

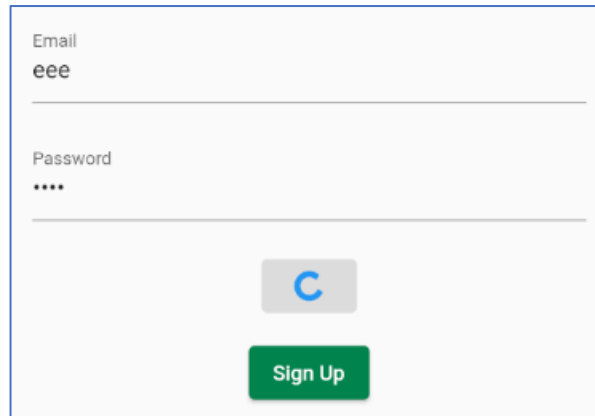


This structure is used to manage information related to student registration, student schedules, student exam results and student lecture information through the files `studentEnrollment.dart`, `studentScheduled.dart`, `studentExamResult.dart` and `lectureStudentInfo.dart`.



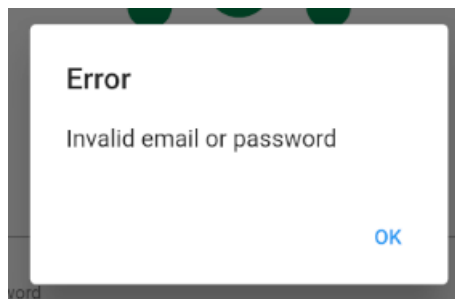
This structure is employed for handling sign-up functionality in `signup.dart`, managing login operations in `login.dart`, and handling other form-related functionalities in various coding files.

## Flutter widget and features adopted in the application

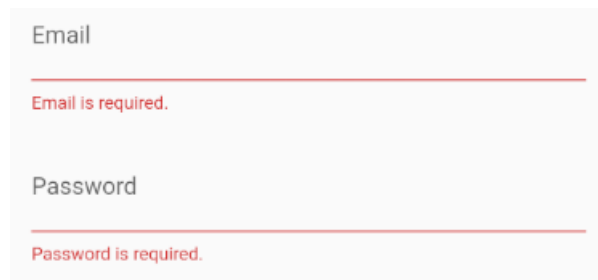


The image shows a login form with two input fields: "Email" containing "eee" and "Password" containing four dots. Below the password field is a grey button with a blue circular progress indicator. At the bottom is a green button labeled "Sign Up".

The `CircularProgressIndicator` is displayed inside the "Login" button when the user is in the process of logging in. This process wants to determine whether the user is a student or a lecturer.



An `AlertDialog` is shown in the event of an error, with an OK button to dismiss the dialog. The `_showErrorDialog` function is responsible for displaying error messages. As shown in the picture above, the message shows that the email or password used is in the database but not with each other.



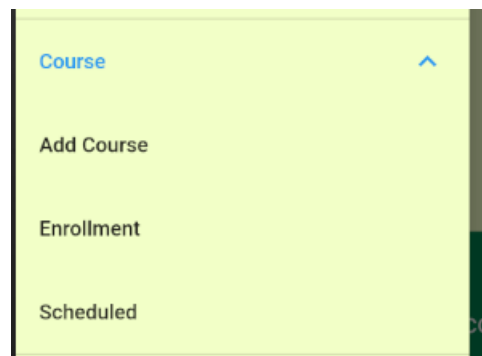
Email

Email is required.

Password

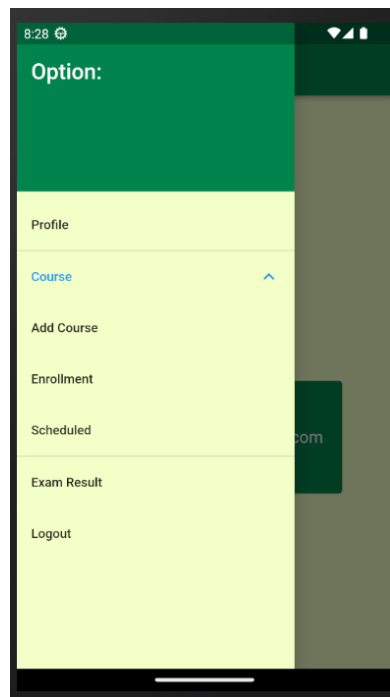
Password is required.

Form validation is implemented via the TextFormField widget for email and password input, incorporating a validation function to ensure that the input is not empty. Additionally, there is a verification mechanism to verify that the phone number entered does not contain letters. This authentication feature is used across registration forms, logins and input of student information when registering for a course.



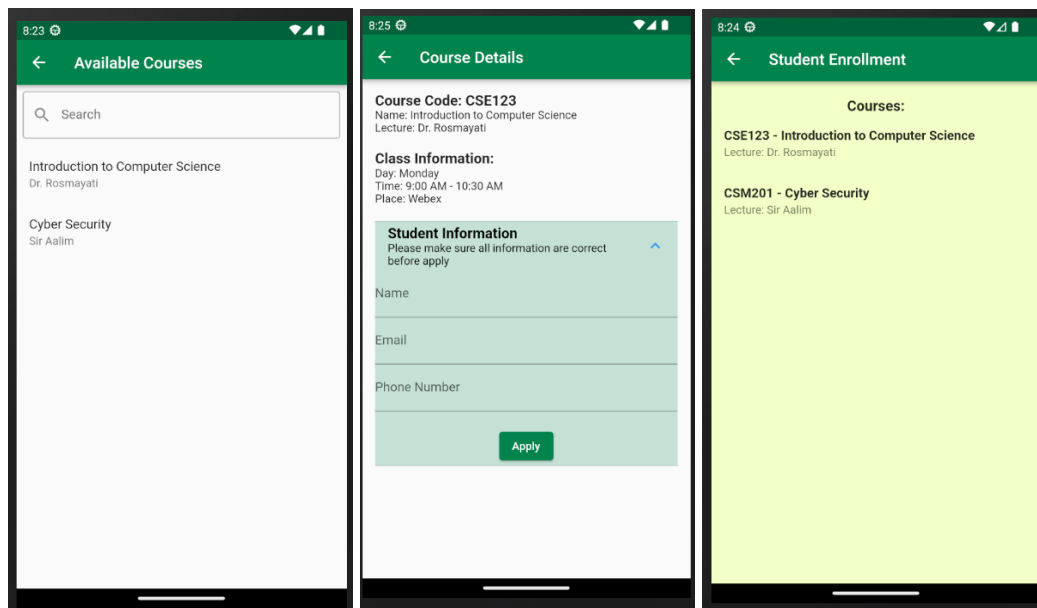
The ExpansionTile widget is used to create expandable tiles to display added, registered and scheduled courses. this is very helpful because all three are under the course category.

## Sample of Interface with the explanation



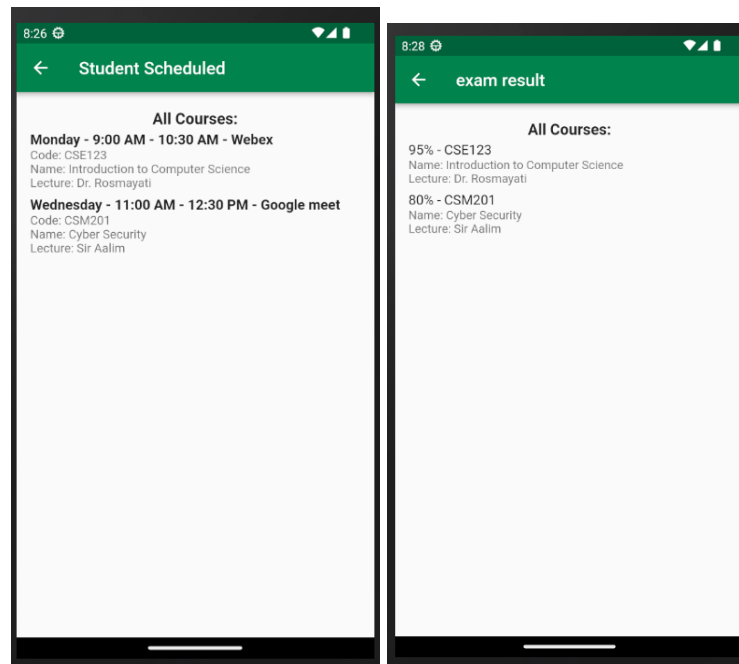
In the Flutter project, there is a side drawer that contains several options such as Profile, Courses, Exam Results, and Log Out. The "Course" option is implemented using an ExpansionTile, which when clicked, will show three additional button like Add Course, Registration, and Schedule. Each of these five buttons (profile, add course, registration, schedule, course, exam results) will navigate the user to the appropriate page. Meanwhile, the Log Out option will allow users to log out of their profile, and redirect them back to the login page.





On the left, there is a view of available courses that allows students to explore and search for specific courses. When users click on a particular course, they will be directed to another page, as pictured in the middle show. This page provides comprehensive details about the selected course, including code, name, lecturer, schedule (day and time), and location.

Below the information, there is a section for student to field student information, student need to make sure all information are right before proceeding. After clicking the "Apply" button, the registration status is updated and the newly applied course will be displayed at enrollment (as shown in the image on the right). At enrollment, if the student want to drop the course, they can easily do it by double click.



The schedule (image on the left) is organized by day, presenting information such as day, time, and location as headings. Meanwhile (image on the right), there is a picture showing the exam results. Only courses that have been applied for by students will be visible in the schedule and exam result.

If the lecturer has not registered the results, the status will be shown as "null". However, if the results have been registered, the result will be shown and will be sorted from highest to lowest. All timetable information and exam results are sourced from Firebase, ensuring that students can only access the data and the process of entering exam results is a task reserved only for lecturers.

## **Conclusion**

The e-course application inspired by Universiti Malaysia Terengganu's MyNemo and the MC plus application offers solutions for students and lecturers in e-course management. With a focus on ease of access and efficient management, this application allows students to register for courses, manage schedules, and monitor exam results without being bound by location or time. Key features providing a seamless experience for students throughout their academic process.

In addition, this application provides support to lecturers by providing tools to monitor registered students and deal with technical problems quickly. The facility to remove students from the course manually and the coordination of exam results marks easily demonstrate the ability of this application in coordinating the needs of students and lecturers. With a simple and easy-to-understand interface design, as well as the efficient use of databases, this e-course application has a positive effect in improving the course management experience for students and lecturers at the university.

## Reference

- ChatGPT. (2024) from Openai.com <https://chat.openai.com>
- MyNemo:Portal Rasmi Warga UMT. (2024). from Umt.edu.my website <https://mynemo.umat.edu.my/>
- Sim. (2014). MCPLUS APPLICATION  
<https://play.google.com/store/apps/details?id=com.simitgroup.mathclinic&hl=en&gl=US>
- CodingZest. (2022). How to Fetch Data From Firebase Realtime Database Flutter | Flutter Firebase Database CRUD 2022 [YouTube Video].  
<https://www.youtube.com/watch?v=zn2vHRbWszY>
- HeyFlutter.com. (2022). Flutter Firebase CRUD (Create, Read, Update, Delete) [YouTube Video]. [https://www.youtube.com/watch?v=ErP\\_xomHKTW](https://www.youtube.com/watch?v=ErP_xomHKTW)
- dbestech. (2021). Flutter App Development | Flutter EBook App UI With Audio Player | Flutter UI Tutorial [YouTube Video].  
<https://www.youtube.com/watch?v=bt61-hktOrc&list=PLgmBguE-4f2xzeOuxoNc-hrSmIXR1Rdjp>
- Dr Mohamad Nor Bin Hassan (2023) Shopping list project
- Dr Mohamad Nor Bin Hassan (2023) expense project
- Read and Write Data | Firebase Realtime Database. (2023)  
<https://firebase.google.com/docs/database/flutter/read-and-write>
- web\_page\_person. (2020, February 3). Flutter Login/Sign-up Screen - Example - TutorialKart. <https://www.tutorialkart.com/flutter/flutter-login-screen/#gsc.tab=0>
- Basic widgets. (2024). Flutter.dev <https://docs.flutter.dev/ui/widgets/basics>

**GitHub**

[https://github.com/srabiatal/s62651\\_project2.git](https://github.com/srabiatal/s62651_project2.git)