UNIVERSITY MALAYSIA TERENGGANU
FACULTY OF OCEAN ENGINEERING TECHNOLOGY & INFORMATICS

FRAMEWORK-BASED MOBILE APPLICATION DEVELOPMENT

(CSM3114)

Title of the Project:
School Management App

Prepared by:

Name	No. Matric
Nurizazi Binti Zulkifli	S61981

Prepared for: Dr. Mohamad Nor Bin Hassan

[Mobile Computing (KMA)]
SEMESTER 1 2023/2024

Table of contents

1.1 Executive Summary

The unique and comprehensive School Management App is redefining the field of educational administration. It is authorized access to vital data is ensured the app's safe and intuitive login process. The administrators are empowered by the expedited processes for adding, updating and monitoring profiles provided by teacher and student management functionalities. All thing considered the School Management App is a versatile and all-inclusive tool that improves communication efficiency and transparency in educational institutions for a more effective and student-focused administration.

The goal of the application is to use the multifunctional single database school management app to assist independent and foreign schools in increasing the productivity, accountability and communication. The cutting-edge software is designed to make school administration as simple as possible. The data will be take seriously because it is very important to protect sensitive data in the digital era now. Modern technology is used by the school management app to ensure that all data is secure and complies with the industry security requirements.

The app is an extensive and effective system that smoothly links all parties involved in the school community. It is an ideal option for any educational institutions because it was created with unique requirements for schools in mind and provides wealth of features and functionality. Furthermore, the apps will make sure all the records will be safely kept and readily available whenever needed and the instructors may monitors the students' progress in a variety of ways, including academic performance. The app will keep the record of all correspondence with teachers and students and handles student hiring, arrange records and evaluation summary.
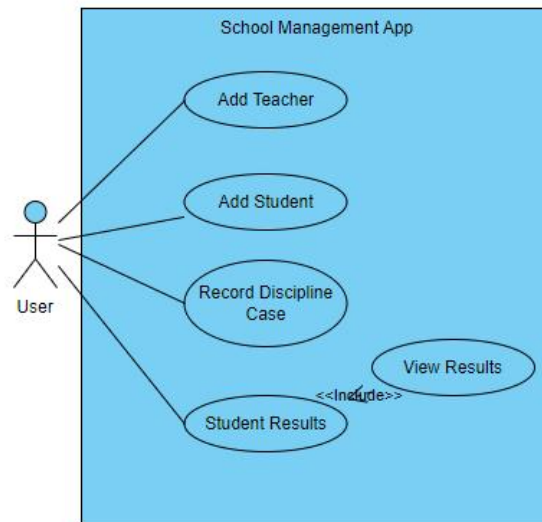
1.2 Use Case

Figure 1 - Use Case

The goal of the School Management App is to improve administrative efficiency in educational establishments by offering administrators, teachers and students a comprehensive and easy-to-use platform. From the use case above, there is one user and the app's key features that have been provided is add teacher, add student, student results, record discipline case and view results.

The user which is can be as school administrator or teachers can play a role as an actor to provide the information inside the apps. The school administrator can add teacher, add student , record the discipline case and record the student results. While, the teachers can view results to track and evaluate student progress and monitor the development of each individual student.

Furthermore, the record discipline case can be enter by both user to add, update or delete the information. This thorough log contributes to the upkeep of a secure and orderly learning environment. In today's educational settings, the school management app is an essential tool to helps teachers and school administrator to make decisions by managing data, communicating effectively and being transparent.
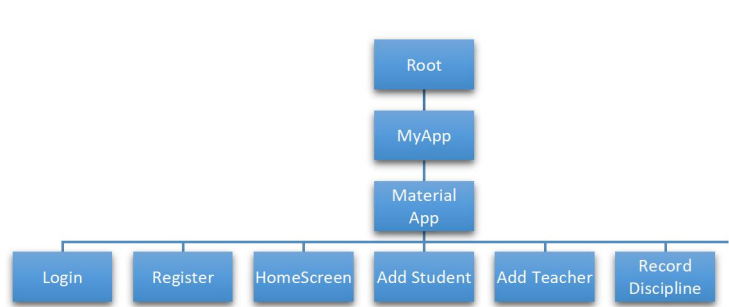
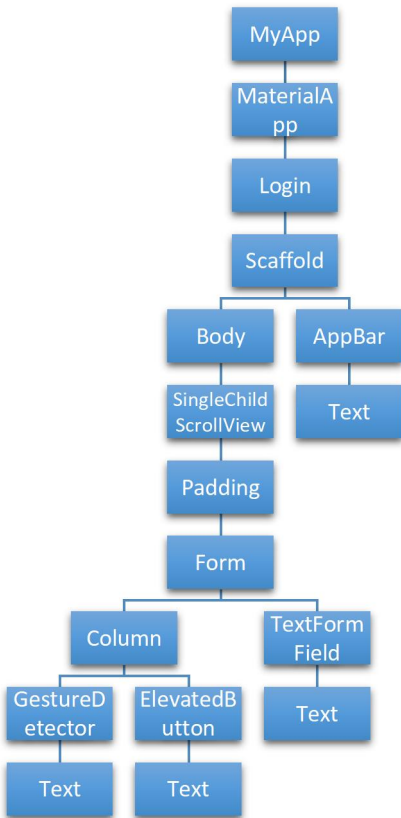## 1.3 Common structure of Tree Widgets
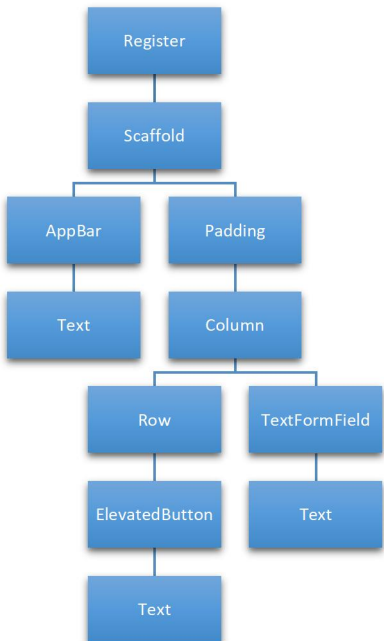

Figure 1 - My App

Figure 2 - Login

Figure 3 - Register

HomeScreen → Scaffold → AppBar → Text; Scaffold → Drawer → Container → ListView → DraweHeader → Text; Scaffold → Container → BoxDecoration → AssetImage; Container → SingleChildScrollView → Column → CarouselSlider; Container → Padding → Row → ListTile → Text
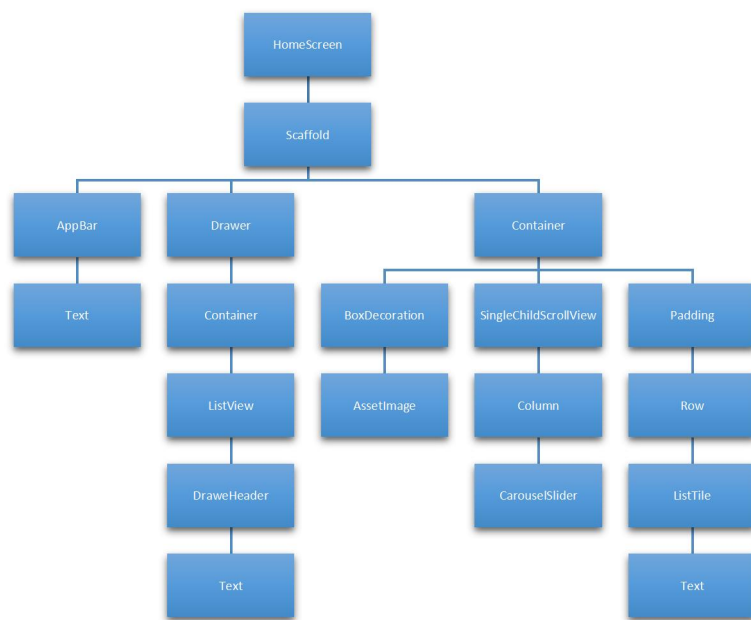
Figure 4 - HomeScreen

Figure 5 shown the widget tree of Add Student and was also implemented in Add Teacher, Record Discipline Case and Student Results
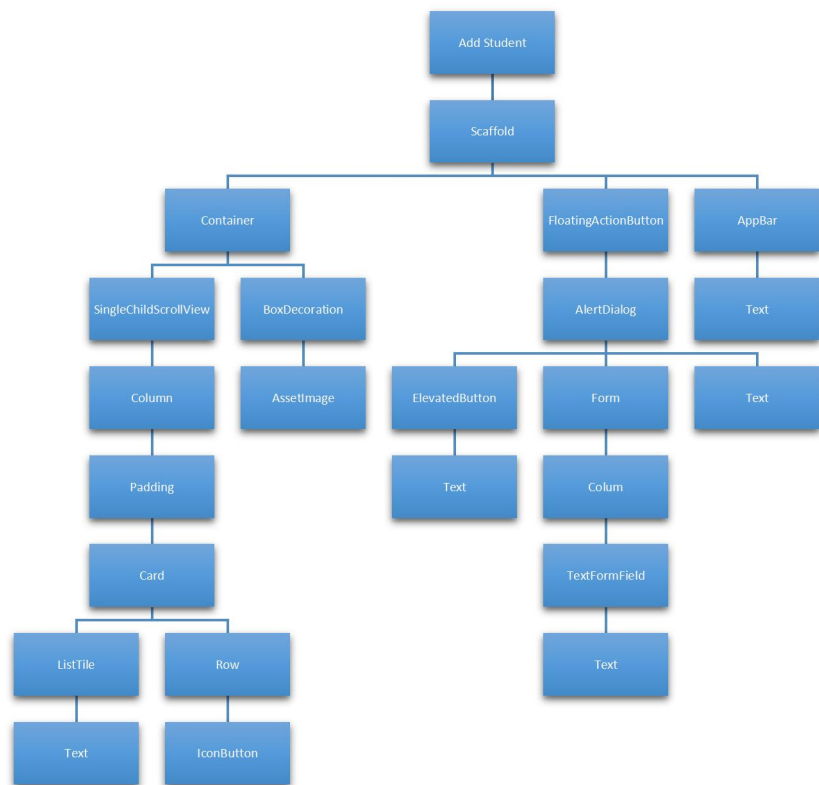

Add Student → Scaffold → Container → SingleChildScrollView → Column → Padding → Card → ListTile → Text; Card → Row → IconButton; Container → BoxDecoration → AssetImage; Scaffold → FloatingActionButton → AlertDialog → ElevatedButton → Text; AlertDialog → Form → Colum → TextFormField → Text; AlertDialog → Text; Scaffold → AppBar → Text

Figure 5 - Add Student

6

1.4 Flutter Features and Widgets

The efficiency, transparency and efficacy of the School Management App in educational administration are all influenced by the aspects taken together. By integrating secure login procedures, streamlined teacher and student administration, academic result tracking and disciplined case recording and accessible student outcomes, the app intends to better overall school management operations and contribute the success of the school.

Login mechanism function as a secure gateway, guaranteeing that access to the app is restricted to authorized persons only, such as school administrators and teachers. To protect the sensitive data, it uses the techniques likes password protection, if needed, multifactor authentication. For the add teacher, the tools gives school administrators the ability to effectively manage teacher profiles and allows for the addition, updating and deleting data. The unified database contains correct and current instructor information thanks to the expedited approach.
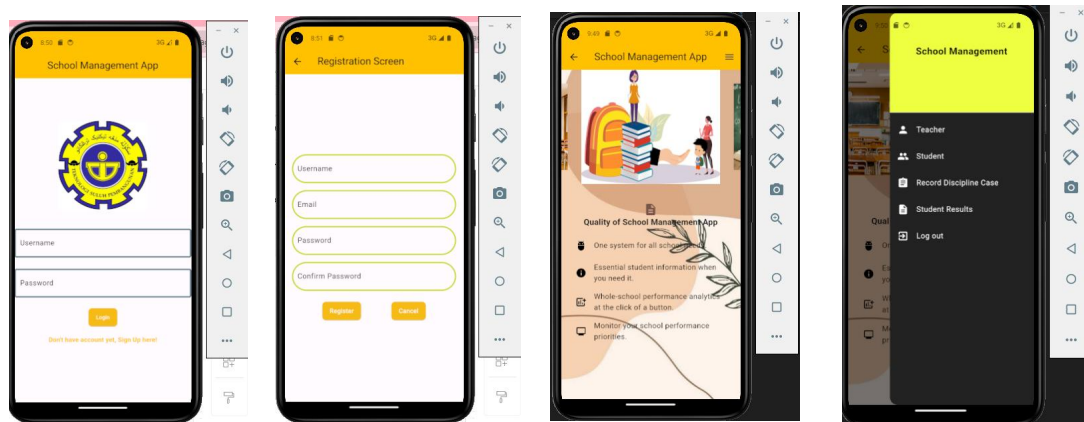
Next, the add student function streamlines the procedures of organizing student data. Student profiles, which is include the student ID and enrollment class, are inputtable and updateable by teachers and school administrators. This extensive student administration features helps create a database that is easily accessible and well-organized. Furthermore, the records discipline case can be use to document specifics of disciplinary incidents. Details include with the id of the student who was involved, an account of what happened and what was action taken. This contribute to a secure and orderly learning environment by helping to keep track of disciplinary actions.

Last features is students results and view results which is to track and evaluate students performance, educators and school administrators can use the view results function. It offers information on averages marks and it is data-driven strategy aids in making well-informed decisions for bettering instruction.

The School Management App's primary user interface is the home screen. I have used the carousel slider in order to manually control the position of the image that I have added in the screen. There is many carousel slider methods that can be used to make the screen more appeal and interesting. The version of the carousel slider that I have used is 4.0.0. Besides, for the login page, I have include the gesture detector that attempts to recognize the gestures that correlate to its non-null callbacks. Gesture detector was used for the onTap text, so there is no need to add a button.

Furthermore, for the display listview of the add student, add teacher, record discipline case and student results, there is a class which is NeverScrollableScrollPhysics that prevents scrolling by the user. There is also other ScrollPhysics class that can be used when making a coding. For the drawer, I have been used buildDrawerItem and NavigateToPage to make the drawer more interesting and appeal. In order to make it simpler to add items to the navigation drawer in a legible and maintained way, the buildDrawerItem is used to construct uniform and styled drawer items with a title, icon and tap action. While the NavigateToPage is where the user tap the title, it will go to the next screen that have been navigate to. In summary, I have used many others basic widgets which is form, TextFormField, Scaffold, AppBar, icons, showDialog and others.
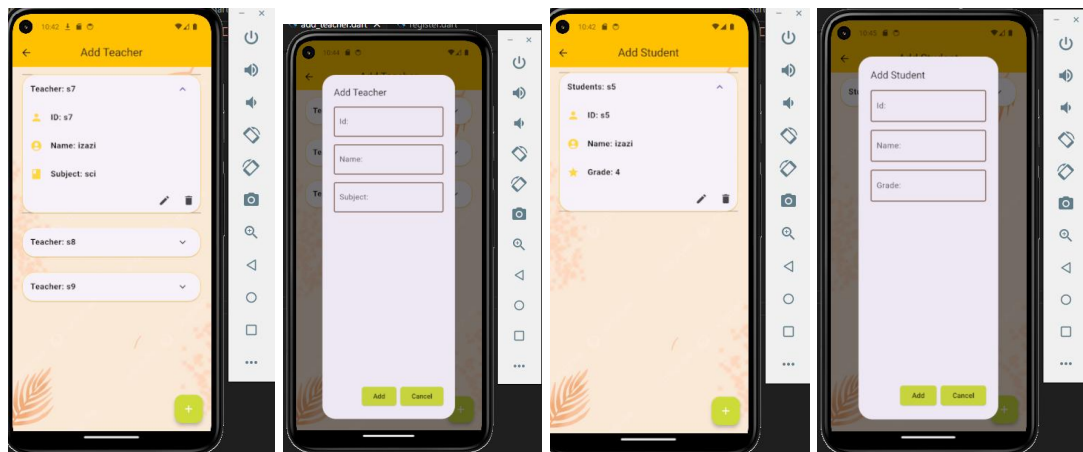
1.5 Sample of Interface

     Login page, where the user key in their username and password to enter the home screen. There is a logo of the school and for the user that did not have the account, they can click the sign up link.

To enter the main screen, user need to log in first and for the log in, they need to have their own account. So, figure 3 is register page that user need to fill in the information, which is username, email, password and confirm password. If the user did not completely fill in the form, the system will show a dialog 'Please fill in the form'. There is two button which is register and cancel.
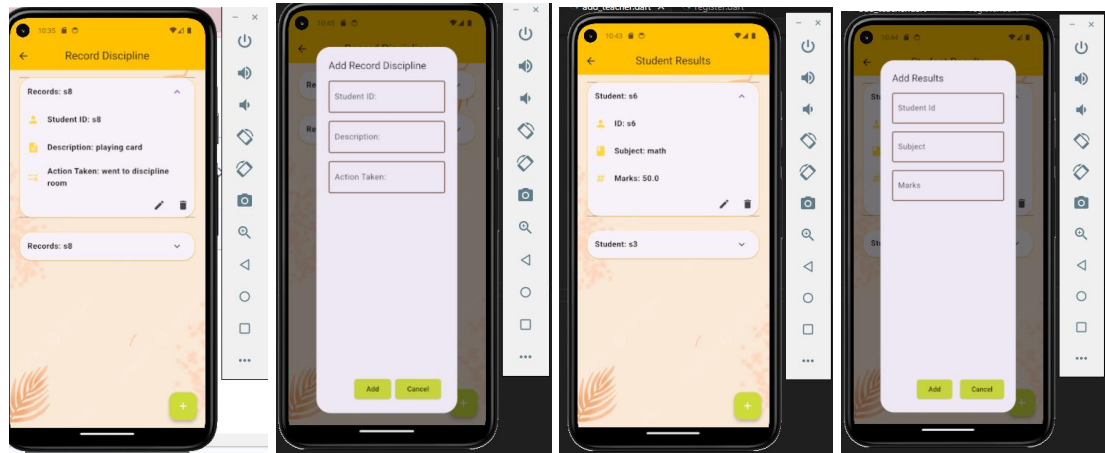
Next, the figure shows the home screen, that have slide image and description about the apps. The purpose of the screen is to make the User Interface more interesting and at the same time looks so professional as the school management app.

The drawer is where the main features was added, which is teacher, student, record discipline case and student results and also log out. Log out will go to the login page after clicked. While, the other features will go to the their page to fill the information or view the information that has been key in.

Add teacher is where the teacher was added by the school administrator and act as user to added and view teachers. There will be id, name and subject that need to be key in. The apps also provided the update and delete information if they need any changes or need to be delete. The background is added to make the list more interesting and at the same time looks professional as in school.

Add Student is where the student was added by the school administrator and teachers to added and view students. There will be id, name and grade that need to be key in. The apps also provided the update and delete information if they need any changes or need to be delete. The background is added to make the list more interesting and at the same time looks professional as in school.

Record discipline case is where the record of discipline students been recorded and been viewed by the teachers and school administrator. If there any changes, they can update or delete the records. There is three things that need to be key in which is the student id, description and action taken.

Next, the student results is where the results was added and the teachers can viewed. It is same as the add teacher, add student and record discipline case function which is, they can update and delete the information if have any changes. They need to key in the student id, subject and marks to be kept in the database.

1.6 Conclusion

In summary, the school management app proves to be a game-changing and essential resource for educational establishments. The software meets the many needs of contemporary schools by slickly integrating features including safe login processes, effective teacher and student management, academic result tracking, discipline records management and strong communication hubs. The administrators, teachers and students may all easily manage complex educational processes thanks to its versatility and user-friendly design.Furthermore, the school management app is a real-time updates and analytics, is at the forefront of creative solutions, allowing educational institutions to concentrate on the primary goal of giving students a supportive and productive learning environment. All thing considered the software is a catalyst for improvement, encouraging, effectiveness, responsibility and quality in school administration. It is a revolutionary development in the field of educational administration since it provides a comprehensive and flexible response to the many problems that contemporary schools encounter.In addition, to streamlining administrative procedures, the app promotes data-driven decision making and real-time updates.

1.7 Reference

1. *Engage: Discover the Best School Management Information System*. (n.d.). Education Horizons. Retrieved January 12, 2024, from https://educationhorizons.com/solutions/engage/?utm_source=google&utm_medium=cpc&utm_campaign=Engage&utm_source=google&utm_medium=cpc&utm_term=school%20management%20app&gclid=EAIaIQobChMIxpnvzYTXgwMVS6JmAh1cGwliEAAYASAAEgIxhfD_BwE

2. *carousel_slider | Flutter Package*. (n.d.). Dart Packages. https://pub.dev/packages/carousel_slider

3. *NeverScrollableScrollPhysics class - widgets library - Dart API*. (n.d.). Api.flutter.dev. Retrieved January 13, 2024, from https://api.flutter.dev/flutter/widgets/NeverScrollableScrollPhysics-class.html

4. *GestureDetector class - widgets library - Dart API*. (n.d.). Api.flutter.dev. https://api.flutter.dev/flutter/widgets/GestureDetector-class.html