Uqarar: Student Degree Plan Based Upon their Academic Record and Mental Health

SUBMITTED BY

Moin Uddin	FA19/BSCS/016
Sarah Ahmad Pirzada	FA19/BSCS/018

SUPERVISOR
Ms. Iram Hina

DEPARTMENT OF COMPUTER SCIENCE
Sir Syed CASE Institute of Technology
ISLAMABAD

2023

CERTIFICATE OF APPROVAL

This project titled

Uqarar: Student Degree Plan Based Upon their Academic Record and Mental Health

Submitted
for the Degree of
Bachelor of Science in Computer Science

By

Moin Uddin	FA19/BSCS/016
Sarah Ahmad Pirzada	FA19/BSCS/018

Has been approved for

Sir Syed CASE Institute of Technology
ISLAMABAD

Supervisor
Ms. Iram Hina
Lecturer
BSCS

Chairman
Dr. Syed Jawad Hussain
Head of Department
BSCS

DECLARATION

I declare that all material in this report is my own work and no material has previously been submitted and approved for the award of a degree by this or any other university.

Moin Uddin: (FA2019/BSCS/016) Signature: _____

Sarah Ahmad Pirzada: (FA2019/BSCS/018) Signature: _____

I certify that the work in this thesis is carried out and completed under my supervision.

Signature: _____

Ms. Iram Hina

Lecturer

Table of Contents

ACKNOWLEDGEMENT

We would like to begin by expressing our heartfelt gratitude to Allah, the Almighty, for granting us the strength, guidance, and perseverance to complete our Final Year Project. Without Allah's blessings and mercy, this accomplishment would not have been possible.

We would also like to extend our sincere appreciation to our supervisor, Ms. Iram Hina, for her unwavering support, encouragement, and valuable guidance throughout our research journey. Her insightful feedback, timely advice, and constructive criticism played a pivotal role in the successful completion of our Final Year Project.

We are also thankful to the faculty members of the Computer Science department for providing us with a conducive learning environment and the necessary resources to conduct our project.

Last but not least, we owe a debt of gratitude to our families and friends for their unwavering love, support, and motivation throughout our academic journey. Their constant encouragement and belief in us have been the driving force behind our success.

DEDICATION

We dedicate our project to our supervisor, Ms. Iram Hina, who has been a constant source of guidance, support, and inspiration throughout our project's development. Her unwavering commitment to our success, coupled with her extensive knowledge and expertise in the field, has been instrumental in shaping our project's direction and refining our ideas.

Ms. Iram Hina's passion for teaching and commitment to excellence has not only challenged us to strive for our best but has also instilled in us a deep appreciation for the field of Computer Science. We are grateful for the opportunity to have worked under her guidance and for the knowledge and skills gained throughout this project's journey.

We express our sincere gratitude to Ms. Iram Hina for her invaluable contribution to our academic growth and success. This project would not have been possible without her support and mentorship. We dedicate this project to her as a symbol of our admiration, respect, and gratitude.

ABSTRACT

In light of the COVID-19 pandemic and its resulting effects on students' mental health and academic performance, it is imperative to recognize the significance of the relationship between mental health and academic achievement. As such, we present a novel solution to address the issue of compromised academic progress due to mental health concerns in undergraduate students.

Many students struggle to adhere to prescribed course schedules in their curriculum due to mental health issues. For instance, a student may miss a prerequisite course in semester 1, preventing them from taking a required course in semester 2. These situations require a study plan adjustment based on their emotional well-being.

Our project aims to develop a mobile application system that analyzes students' academic and non-academic data to generate the most optimal outcome for their academic progression. Specifically, we focus on the challenge faced by students who are unable to adhere to the prescribed course schedule in their curriculum due to mental health concerns. For instance, a student may be required to complete a prerequisite course in semester 1 to take a course in semester 2. However, if the student misses the prerequisite course in semester 1, they will not be able to take the course in semester 2, thus necessitating a change in their study plan based on their emotional well-being.

In this paper, we propose a mobile/web application that assists students in planning the sequence of remaining courses in their curriculum, while ensuring that the plan is optimized based on their mental health status. To achieve this, we employ a genetic algorithm that considers all pertinent course conditions, including prerequisites or opening semesters of each course, the total credit hour limit of each semester, as well as the estimation of each semester's GPA and emotional health indicators.

We believe that our proposed solution will be of immense benefit to undergraduate students and student counselors. By providing a personalized and optimized academic progression plan, our application can help to alleviate the mental health-related barriers to academic success, thus promoting greater well-being and academic performance among undergraduate students.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1.1. Brief

Given the COVID-19 pandemic and the consequent struggles in life and alternative learning environments, students' mental health and academic success are compromised. The challenges such as social isolation, increased stress, disrupted routines, and the shift to remote learning, have collectively contributed to this compromise. The relationship between mental health and academic achievement is a two-way street and should not be taken lightly by teachers and students alike. Therefore, our Final Year Project (FYP) aims to develop a mobile application system that will analyze students' academic and non-academic data e.g. their mental health, to generate the best possible outcome for them to follow.

Our system comprises two modules, namely a prediction system that analyzes the student's academic data, and a counseling module that employs machine learning algorithms to analyze their mental health. As this unique aspect is often overlooked by teachers, we see our project as a real-time application that universities can use.

To create the recommendation system for the first module, we need to consider the credit hour limit and teachers' availability when providing the best plan for students to complete their courses, including those they may have dropped. In the second module, we will use machine learning algorithms to detect students' mental health, involving data extraction, pre-processing, training Deep Learning classifiers, and detection classification.

The flow of the system is as follows: collecting and preparing student data, interpreting the data, and creating theories about the student's performance, taking necessary actions to meet their needs, and testing those theories by making changes. This process should repeat indefinitely to ensure continuous reflection and evaluation of any changes or improvements. However, certain limitations exist, such as the ever-changing data of students and the course plan changes for every batch, which will also need to be addressed.

Using data thoughtfully to ask questions and obtain insight into student progress is a logical way to monitor continuous improvement and target instruction to the needs of each student. By implementing this strategy, students can experience an enhancement in their overall GPA, while teachers can provide counseling and prioritize students'

mental health. This comprehensive approach will lead to significantly improved outcomes.

1.1.2. Relevance to Course Modules

The proposed mobile application system for optimizing students' academic progress and mental health employs several concepts that are relevant to the course. One such concept is the use of genetic algorithms, which falls under the umbrella of artificial intelligence. By utilizing genetic algorithms, the application can consider all relevant course conditions and generate an optimized academic progression plan. Another relevant concept is emotion recognition, which is a subfield of machine learning. Through the use of emotional health indicators, the application can take into account students' mental health status when creating their academic plan. Additionally, the application is developed using Flutter, a tool for smart application development, allowing for easy and efficient creation of mobile and web applications. Finally, the system utilizes CSV and Firebase management, which are database systems used to store and manage large amounts of data, such as students' academic records and emotional health data. Overall, the proposed mobile application system incorporates several relevant concepts from the course, making it a valuable and practical project.

Table 1 Concept Relevance to Course Module

Concept Used	**Relevance to Course**
Genetic Algorithm	Artificial Intelligence
Emotion Recognition	Deep Learning
Flutter	Smart Application Development
CSV and Firebase management	Database System

1.1.3. Methodology and Software Lifecycle

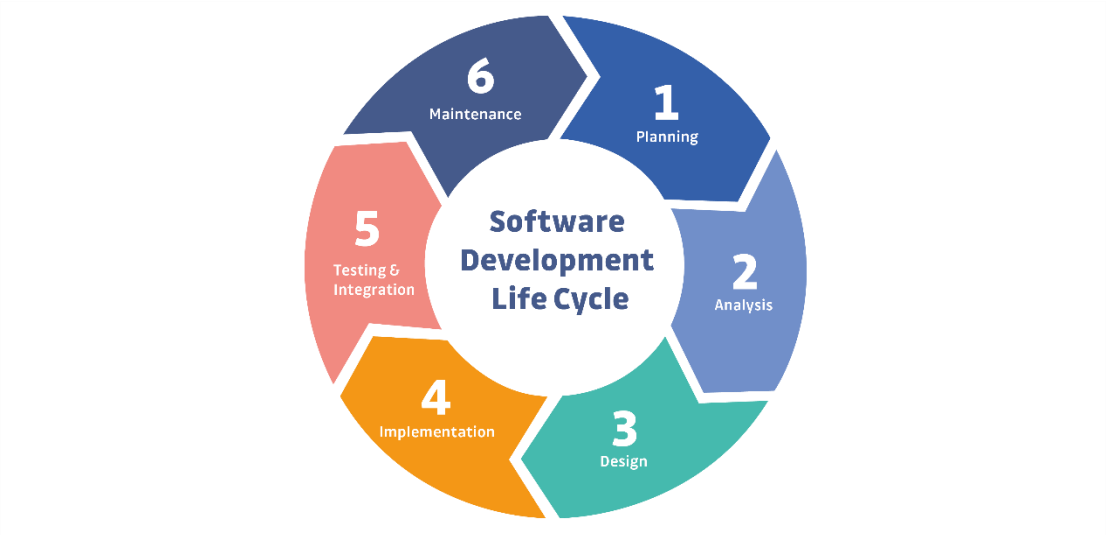


Figure 1 Software Lifecycle

Planning

Our approach to this project is grounded in a systematic and comprehensive strategy, driven by a meticulous sequence of steps designed to ensure its success. At the outset, our commitment to excellence manifests in the form of thorough research. This research endeavor is multifaceted, encompassing a deep exploration of the subject matter, technological landscape, and user expectations. By delving into these areas, we gain insights that inform the subsequent stages of the project, enabling us to make informed decisions and strategic choices.

Central to our strategy is the establishment of clear and well-defined objectives, to optimize academic success, provide mental health support, offer customization, reduce student dropouts, empower counselors, and promote holistic well-being among students. These objectives serve as guiding beacons, aligning our efforts towards a shared vision of what the project aims to achieve. By crystallizing these objectives, we create a roadmap that directs our actions, shapes our priorities, and aids in effective decision-making throughout the project's lifespan. Each objective serves as a yardstick against which we can measure progress and success, ensuring that every effort contributes meaningfully to the overarching goals.

An equally pivotal facet of our strategy involves the meticulous crafting of a project timeline. This timeline acts as a structured framework, delineating the sequence of

tasks, milestones, and deadlines that chart the project's progression. By breaking down the project into manageable segments and assigning timeframes to each, we create a sense of order and accountability. This timeline not only serves as a blueprint for the project's execution but also provides a means to track progress, manage resources, and ensure that the project remains on course.

Crucial to our commitment to quality and reliability is the iterative testing process that the application undergoes. Rigorous testing is undertaken multiple times, with precision and thoroughness, to ensure the utmost accuracy and reliability of the application's functionalities. This iterative testing approach allows us to identify and rectify any issues, fine-tune features, and enhance performance. By subjecting the application to rigorous scrutiny, we build a solid foundation of confidence in its capabilities, ensuring that it delivers on its intended promises.

In essence, our strategy is underpinned by a blend of research, clear objectives, structured timelines, and rigorous testing. This holistic approach not only instills discipline into the project's lifecycle but also positions us to optimize efficiency, maintain focus, and cultivate a final product of exceptional accuracy and value. Each step is an integral component of a larger journey that is driven by a resolute commitment to delivering excellence and achieving the project's ultimate goals.

Analysis

The undertaking of this project necessitates a comprehensive and multifaceted exploration, encompassing a range of factors that interplay in shaping both students' academic progress and their mental well-being. This intricate journey begins with an exhaustive analysis aimed at deciphering the nuanced dynamics that influence these dimensions.

To embark on this endeavor, a crucial foundational step involves delving into the extensive realm of academic literature, particularly focusing on the intersection of mental health and academic attainment in the context of undergraduate students. This entails a meticulous review of existing studies, research papers, and scholarly articles that highlight the intricate interplay between mental health concerns and academic achievements. This in-depth literature analysis serves as the bedrock for understanding the multifaceted nature of this subject, providing valuable insights into the diverse factors that contribute to mental health challenges among students. It involves dissecting the

gamut of variables, encompassing academic stressors, social isolation, anxiety triggers, and other key elements that influence students' emotional well-being.

Moreover, a holistic perspective necessitates a detailed exploration of the academic landscape itself. This involves scrutinizing the various intricacies of course conditions, prerequisites, credit hour constraints, and the opening semesters for each individual course. This meticulous examination ensures a nuanced understanding of the academic requirements that students encounter, illuminating potential stumbling blocks or challenges that might affect their academic progression.

Going beyond the present, historical course data and emotional health indicators are tapped to forecast and estimate the academic landscape's future contours. By analyzing past course records and emotional health trends, it becomes possible to extrapolate estimates for factors like Grade Point Average (GPA) and cumulative GPA for each upcoming semester. This predictive analysis paints a forward-looking picture, providing a glimpse into potential academic trajectories that students might navigate based on historical patterns and emotional well-being indicators.

In essence, this project rests upon an intricate tapestry of analyses that converge to unravel the multifaceted link between academic progress and mental health. By systematically dissecting academic prerequisites, emotional indicators, historical course data, and the complex psychological factors affecting students, a holistic understanding is achieved. This understanding becomes the cornerstone on which our application is built, facilitating the creation of academic progression plans that not only reflect the academic realities but also incorporate the critical dimension of mental well-being, fostering a comprehensive solution that addresses both academic excellence and students' emotional health.

Design

We created a wireframe of the application, which will serve as a blueprint of the application's structure and flow. The wireframe will identify the various functionalities and features that the application will have. Then we focused on creating a visual design for the application. This involved choosing appropriate colors, typography, and icons that align with the application's purpose and target audience. The visual design is simple, clean, and easy to navigate, with a user interface that is intuitive and easy to understand.

Implementation

The implementation phase of our project stands as a pivotal milestone where meticulous planning and creative design converge into concrete reality. In this critical juncture, our focus was channeled towards the integration of all the carefully crafted designs and essential functionalities that were identified as prerequisites for our project's success. This synthesis of form and function was realized through a harmonious synergy of two powerful technologies: Python and Flutter.

In the realm of backend development, Python emerged as the linchpin of our implementation strategy. Renowned for its readability and versatility, Python was instrumental in translating our design blueprints into functional components. With its rich ecosystem of libraries and frameworks, Python facilitated the seamless creation of complex algorithms and logic that underpin the core functionality of our application. Whether it was the meticulous calculation of academic trajectories or the intricate evaluation of mental health indicators, Python's prowess was harnessed to weave these elements into a cohesive tapestry of utility.

On the frontend, Flutter emerged as the visual canvas that transformed our designs into an interactive and aesthetically pleasing user interface. Flutter's cross-platform compatibility allowed us to develop a consistent user experience that transcends device boundaries. Its widget-based architecture facilitated the construction of intuitive and dynamic user interfaces, replete with animations, transitions, and responsive elements that bring our application to life. This framework enabled us to craft an engaging and visually coherent environment that seamlessly incorporates the functionalities we designed.

The symbiotic integration of Python and Flutter encapsulates the essence of our implementation. By harmonizing the backend logic and frontend presentation, we forged a dynamic and harmonious whole. Our meticulous planning and design efforts found their ultimate realization through the adept utilization of these technologies, resulting in an application that not only boasts the functionalities we envisioned but also provides a user-friendly and visually compelling experience. As we navigated the intricacies of implementation, Python and Flutter became the mediums through which our vision transformed into a tangible and impactful reality, serving as a testament to the potential of thoughtful design and meticulous execution.

Testing and Integration

The validation of our project's robustness and functionality rested upon a meticulous and comprehensive testing methodology that spanned multiple phases. This approach was anchored in a commitment to ensuring that each module, as well as the integrated system as a whole, operated seamlessly and in alignment with our intended goals.

The architectural framework of our system was designed to accommodate the diverse testing requirements. The process commenced with the individual assessment of each module, a stage characterized by thorough testing on two distinct platforms: Google Colab and Visual Studio Code. The rationale behind this approach lay in harnessing the strengths of both environments. Google Colab, with its cloud-based capabilities, allowed us to execute and test our modules in an environment that provided necessary resources and computational power. On the other hand, Visual Studio Code offered an agile and robust integrated development environment that facilitated the creation and assessment of these modules within a controlled local setting.

This meticulous individual module testing was pivotal in unearthing potential issues, bugs, or areas requiring optimization. Each module's functionality, from intricate algorithmic calculations to user interface components, underwent scrutiny to ensure that it operated as intended, adhering to design specifications and functional requirements.

Subsequently, the integration phase took center stage, where these thoroughly vetted modules were harmoniously woven into the fabric of our back-end code. The integration was conducted methodically, module by module, with vigilant attention to maintain the integrity of each component's functionality during this process.

This was followed by a critical step that underlined our commitment to a holistic, reliable system: end-to-end testing. At this juncture, the entire system was put to the test, simulating real-world interactions and scenarios to verify that all modules cooperated seamlessly within the architectural framework. This rigorous evaluation encompassed the complete application, ensuring that interdependencies, data flows, and interactions unfolded without glitches.

By conducting these multi-tiered testing cycles within our chosen architectural framework, we affirmed the stability and dependability of our application. This holistic approach, transitioning from individual module validation to comprehensive system assessment, was rooted in our endeavor to assure the quality, performance, and reliability of every aspect of the application. Through this methodical testing process, we ascertained that our meticulously designed modules seamlessly converged within the architectural framework to fulfill our intended objectives, ultimately producing an application that was both resilient and poised to deliver an impactful user experience.

Maintenance

At the heart of our product's design philosophy is a profound emphasis on self-sufficiency and minimizing the need for ongoing maintenance. We've meticulously crafted the architecture and codebase with the overarching goal of creating a solution that operates seamlessly, ensuring a robust and reliable user experience from the outset. This guiding principle centers on the anticipation and mitigation of potential issues before they arise, thereby reducing the necessity for frequent maintenance interventions.

While our intention is to create an application that operates autonomously, we acknowledge that the reality of software development entails the possibility of unforeseen errors or glitches. In the event that such an anomaly does occur, we are primed to respond swiftly and effectively. Our commitment to delivering a dependable product extends to our approach to rectifying any issues. In such cases, we will embark on a targeted process to identify the root cause of the problem and devise a tailored solution. This solution-oriented approach reflects our dedication to ensuring a seamless user experience, even when challenges arise.

This proactive stance towards addressing issues aligns with our commitment to user satisfaction and product excellence. By swiftly addressing any errors that might surface, we ensure that our product consistently delivers on its intended promise. Our comprehensive approach to quality assurance, coupled with our readiness to provide swift resolutions, underscores our commitment to not only creating a self-sufficient product but also standing behind its reliability over time.

CHAPTER 2: PROBLEM DEFINITION

2.1.1. *Literature Review*

The relationship between mental health and academic achievement has been a subject of significant research interest over the years. Studies, such as the one conducted by Thornicroft et al. in 2011, have shed light on the critical relationship between mental health and academic performance. Their findings underscore how mental illness can exert a detrimental influence on a student's ability to excel academically. Additionally, they emphasize that the stress associated with academic demands can, in turn, lead to the development or exacerbation of mental health problems.

Further supporting this linkage, research by Hunt and Eisenberg in 2010 has demonstrated that students grappling with mental health issues face an increased risk of dropping out of college. These findings underscore the real-world consequences of unaddressed mental health concerns on educational pursuits.

Given these insights, it is evident that focusing solely on academic factors is insufficient to ensure student success. The dynamic interaction between academic and non-academic elements of a student's life must be considered. Our project takes up this imperative challenge by aiming to develop an accessible and practical tool that empowers student counselors. This tool facilitates the creation of customized academic plans for multiple students, integrating both academic and non-academic considerations, thus addressing the complex relationship between mental health and academic achievement. The objective is to provide a holistic support system for students, promoting their well-being and academic progress while reducing the risk of attrition.

2.1.2. Problem Statement

The problem at hand is the impact of mental health concerns on academic success and vice versa. Students often struggle to follow their prescribed course schedules due to mental health issues, hindering their academic progress. To address this challenge, we aim to develop a mobile application that analyzes both academic and non-academic data to optimize students' academic progression. This includes tailoring study plans based on their mental health status. The objective is to alleviate the barriers created by

mental health issues and foster greater well-being and academic performance among undergraduate student.

CHAPTER 3: REQUIREMENT ANALYSIS

3.1.1. *Use Case Diagrams*

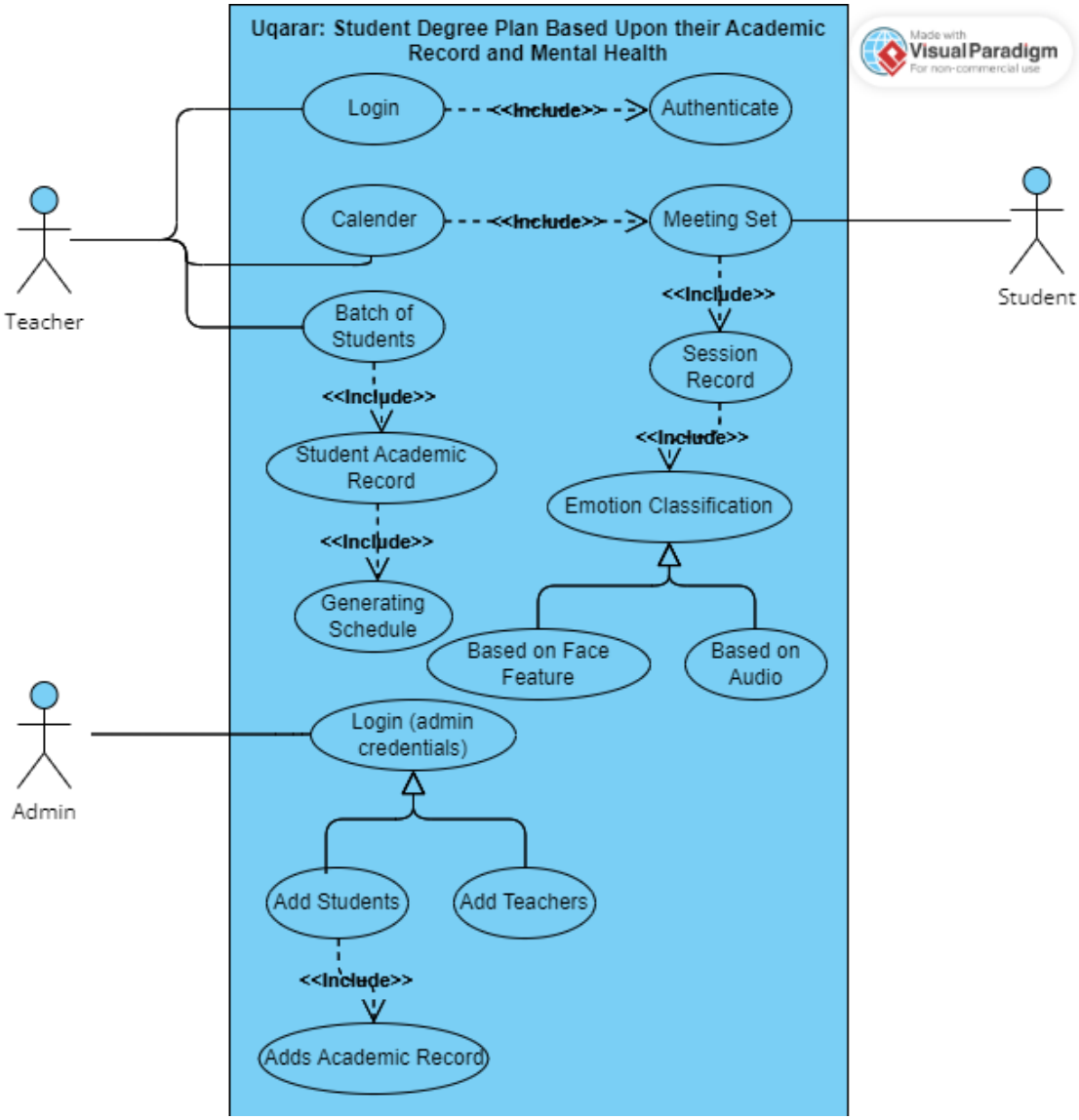


Figure 2 Use Case Diagram

3.1.2. Use case name and identifier (Use Case Narrative)

Login

1.	Use Case Name: Login	
2.	Implementation Priority: 1	
3.	Actors: Teachers	
4.	Summary: Teacher logs into his\her account	
5.	Pre-condition:	
6.	Post-condition: Teacher is allowed into the app.	
7.	Extend: -	
8.	Include: Authentication	
9.	Normal Course of Events:	
	System Actor Action	System Response
	1. Teacher types in his\her login credentials	2. System send the response to authenticator.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 2 Login

Authenticate

1.	*Use Case Name:* Authenticate
2.	Implementation Priority: 1
3.	Actors:
4.	Summary: Back end of our login system
5.	Pre-condition: Worker should be logged in.
6.	Post-condition:
7.	Extend: -
8.	Include: -

9.	Normal Course of Events:	
	System Actor Action	System Response
		1. System lets the teacher login if the login credential is correct.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 3 Authenticate

Calendar

1.	*Use Case Name:* Calendar	
2.	Implementation Priority: 2	
3.	Actors: Teacher	
4.	Summary: Teacher can see his\her timeline if he\her have a meeting set with any student or not	
5.	Pre-condition: Teacher should be logged in.	
6.	Post-condition:	
7.	Extend: -	
8.	Include: - Meeting Set	
9.	Normal Course of Events:	
	System Actor Action	System Response
	2. Teacher clicks on notification. 4. Teacher taps on Close.	1. System sends notification of message. 3. System opens application and shows the reminder in a dialog box with Close and Reply button. 5. System closes pop up.
10.	Alternative Path:	

11.	Exception: Internet not working.

Table 4 Calendar

Meeting Set

1.	*Use Case Name:* Meeting Set	
2.	Implementation Priority: 2	
3.	Actors: Teacher	
4.	Summary: Teacher can set meeting with any student which comes under his\her umbrella	
5.	Pre-condition: Teacher should be logged in.	
6.	Post-condition:	
7.	Extend: -	
8.	Include: - Session Record	
9.	Normal Course of Events:	
	System Actor Action	System Response
	1. Teacher set a time and date for his\her meeting with the selected student.	2. System sends a notification to the student.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 5 Meeting

Batch of Students

1.	*Use Case Name:* Batch of Students
2.	Implementation Priority: 1
3.	Actors: Teacher
4.	Summary: In the dashboard of the screen the teacher sees what batches are under him\her and she can access each batch individually.
5.	Pre-condition: Teacher should be logged in.

6.	Post-condition:	
7.	Extend: -	
8.	Include: - Student Academic Record	
9.	Normal Course of Events:	
	System Actor Action	System Response
	2. Teacher can access each batch one by one.	1. System show all the batches under the teacher.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 6 Batch of Students

Student Academic Record

1.	*Use Case Name:* Student Academic Record	
2.	Implementation Priority: 2	
3.	Actors: Teacher	
4.	Summary: Teacher can see his\her timeline if he\her have a meeting set with any student or not	
5.	Pre-condition: Teacher should have selected a Batch of Students	
6.	Post-condition:	
7.	Extend: -	
8.	Include: - Generating Schedule	
9.	Normal Course of Events:	
	System Actor Action	System Response
	1. Teacher can see the list of students in the batch. 2. Teacher can now select each student to check their academic record	3. System will show the whole academic record of the student.
10.	Alternative Path:	

11.	Exception: Internet not working.

Table 7 Student Academic Record

Session Record

1.	*Use Case Name:* Session Record	
2.	Implementation Priority: 2	
3.	Actors: Teacher	
4.	Summary: Teacher when the meeting start can record the student conversation.	
5.	Pre-condition: Meeting should be started	
6.	Post-condition:	
7.	Extend: -	
8.	Include: - Emotion Classification	
9.	Normal Course of Events:	
	System Actor Action	System Response
	1. Teacher Starts recording the meeting,	2. System starts to store the video in the database.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 8 Session Record

Emotion Classification

1.	*Use Case Name:* Emotion Classification
2.	Implementation Priority: 2
3.	Actors:
4.	Summary: After the meeting is recorded, we do apply our Deep Learning algorithm so that recording can be of that student use.
5.	Pre-condition: Meeting should have concluded.
6.	Post-condition:
7.	Extend: -

8.	Include: - Generating Schedule	
9.	Normal Course of Events:	
	System Actor Action	System Response
		1. System first separates the video into two forms frames and audio. 2. First the frames and audio go under processing so we can apply our algorithm respectively. 3. Each frame and audio give us an output which we send it to our next process.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 9 Emotion Classification

Generating Schedule

1.	*Use Case Name:* Generating Schedule	
2.	Implementation Priority: 2	
3.	Actors:	
4.	Summary: From the data of student academic record and his mental state during the meeting session we generate a schedule for our student to follow.	
5.	Pre-condition:	
6.	Post-condition:	
7.	Extend: -	
8.	Include: -	
9.	Normal Course of Events:	
	System Actor Action	System Response
	2. Teacher can check it and can-do manual changes as well if he\her see fit. 3. Then teacher can share this new schedule with the student.	1. System generates a new timeline for the student to follow.

10.	Alternative Path:
11.	Exception: Internet not working.

Table 10 Generating Schedule

Add Students

1.	*Use Case Name:* Add Students	
2.	Implementation Priority: 1	
3.	Actors: Admin	
4.	Summary: Admin only has the access to add students in the database which then can be later shown to the teachers.	
5.	Pre-condition: Admin should have credentials	
6.	Post-condition:	
7.	Extend: -	
8.	Include: -	
9.	Normal Course of Events:	
	System Actor Action	System Response
	2. Teacher clicks on add to add students with their information.	1. Database is updated with students.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 11 Add Students

Add Teacher

1.	*Use Case Name:* Add Teacher
2.	Implementation Priority: 1
3.	Actors: Admin
4.	Summary: Admin only has access to add teacher in the system as well assign them to a batch where the teachers can see student record.

5.	Pre-condition: Admin should have his credentials.	
6.	Post-condition:	
7.	Extend: -	
8.	Include: -	
9.	Normal Course of Events:	
	System Actor Action	System Response
	2. Admin clicks on add to add teacher. 3. Admin assigns teacher to a batch	1. Data is updated in the database system.
10.	Alternative Path:	
11.	Exception: Internet not working.	

Table 12 Add Teachers

Add Academic Record

1.	*Use Case Name:* Add Academic Record	
2.	Implementation Priority: 1	
3.	Actors: Admin	
4.	Summary: Admin adds the students' academic record of each student so it can be shown to teachers and. be used	
5.	Pre-condition: Student should be added in the system.	
6.	Post-condition:	
7.	Extend: -	
8.	Include: - Meeting Set	
9.	Normal Course of Events:	
	System Actor Action	System Response

	2. Teacher clicks on add and uploads students ‘academic result.	1. System is updated with students’ information and now can be used in the planner by teachers.
10.	Alternative Path:	
11.	Exception: Student is not enrolled in system.	

Table 13 Add Academic Record

3.1.3. Functional Requirements

- **Teacher Assignments:** Assigning teachers to specific batches or groups of students they are responsible for counseling is a crucial aspect of ensuring effective student support. By aligning teachers with their designated groups, it allows for more personalized and tailored guidance. Teachers can develop a deeper understanding of their students' needs and provide relevant advice and assistance.

- **Up-to-Date Academic Data:** Maintaining current and accurate academic data for each student is essential. This data includes records of their academic performance, courses completed, grades, and other relevant academic information. Timely and precise academic data enables teachers and counselors to make informed decisions and recommendations for students' academic progress.

- **Student Contact Information:** Having access to students' contact information is important for efficient communication. This includes email addresses, phone numbers, and any other preferred means of contact. This information ensures that teachers and counselors can easily reach out to students to schedule meetings, provide updates, or address concerns.

- **Automatic Meeting Input:** Implementing an automated system for scheduling and inputting meetings streamlines the process. It allows teachers and

counselors to efficiently schedule counseling sessions with students. Automation can also send reminders and notifications to both parties, ensuring that meetings occur as planned.

- **Feature Extraction:** Feature extraction typically refers to the process of identifying and extracting relevant characteristics or data points from a larger dataset. In this context, it could involve identifying key features or parameters that influence a student's academic progress and well-being. These features might include attendance, previous academic performance, and emotional well-being indicators.

- **Generating Schedule:** The generation of a personalized academic and counseling schedule is a critical component. This involves creating a plan that optimizes a student's academic progression while taking into account their specific needs, academic requirements, and mental health status. It ensures that students can navigate their academic journey more effectively.

3.1.4. Non-Functional Requirements

- **Performance:** The system must deliver a responsive and efficient performance, ensuring timely access to data and smooth operation.
- **Reliability:** It should offer a highly reliable service, minimizing downtimes and ensuring data accuracy.
- **Efficiency:** The application must be designed for efficiency, utilizing system resources optimally.

Maintainability:

- **Modularity:** The application is structured with a modular design, facilitating swift and easy maintenance. This modularity will support rapid updates and improvements while minimizing potential disruptions to the system.

Usability:

- **Cross-Platform Accessibility:** The application will be accessible both on mobile devices and websites, enhancing usability for teachers. This cross-platform capability ensures easy access from various locations and devices, increasing user convenience.

3.1.5. *Safety Requirements:*

- **Data Disposal:** The system should automatically dispose of meeting records after use since they contain highly personal information related to students. Additionally, student academic data must be accessible but stored in an encrypted format to ensure data security and privacy.

CHAPTER 4: METHODOLOGY

The application is developed using the Flutter framework, which is an open-source mobile application development tool created by Google.

The development of the prediction system involved the use of a machine learning algorithm known as Genetic Algorithm. The system analyzed academic data to generate personalized recommendations for students on how to complete their courses. The recommendation system was designed to consider various factors such as credit hour limits and teacher availability to create a feasible and optimal plan for each student. To ensure the accuracy of the recommendation system, a large dataset of academic data was used to train the machine learning algorithms.

The development of the counseling module involved the use of advanced machine learning algorithms, specifically a convolutional neural network (CNN) such as VGG 16. The module was designed to analyze students' mental health by extracting, preprocessing, training, and classifying data using the CNN. To ensure the accuracy of the counseling module, a large dataset of student mental health data was used to train the CNN. The counseling module provides personalized recommendations for each student based on their emotional status and academic data.

Both modules were integrated into a single mobile application system, which was designed to be user-friendly, intuitive, and accessible to both students and student counselors. The system was hosted on a cloud-based platform to ensure scalability, security, and reliability.

In addition, a detailed user manual was created that outlines the system's features and functionalities, installation instructions, and troubleshooting guidelines. A comprehensive test report was also produced, including the testing methodology, test cases, and results, to ensure the application's quality and reliability. The methodology employed in the development of the final deliverable aimed to create an efficient and effective mobile application system that benefits both students and student counselors.

4.1 Workflow Model

1. The first step in developing a comprehensive student support program is to collect academic data for all students, with a particular focus on those who require assistance in improving their academic performance. This may include data such as grades, test scores, scheduled appointments with student counselor and other relevant information, obtained from school records or through input from teachers.

2. The next step is to schedule individual meetings with students who require support. During these sessions, specific questions are asked to evaluate the student's mental health performance. This information is critical in developing a comprehensive support plan that addresses not only the student's academic needs but also their overall well-being.

3. Once academic and mental health data have been collected, the next step is to combine this information to create a comprehensive support plan using a genetic algorithm. This algorithm can analyze the data and generate a personalized plan that addresses the student's academic and mental health needs. This plan may include personalized goals, study schedules, and other accommodations designed to help the student achieve their full potential.

4. The final step is to generate a planner for the student to follow, designed to achieve the best possible results. The planner may include personalized goals, study schedules, and any necessary accommodations to help the student succeed. This planner can be used by both the student and teacher to monitor progress and make any necessary adjustments. The goal of this comprehensive support program is to provide students with the resources and support they need to succeed academically and thrive personally.

1.1. Process Flow Representation

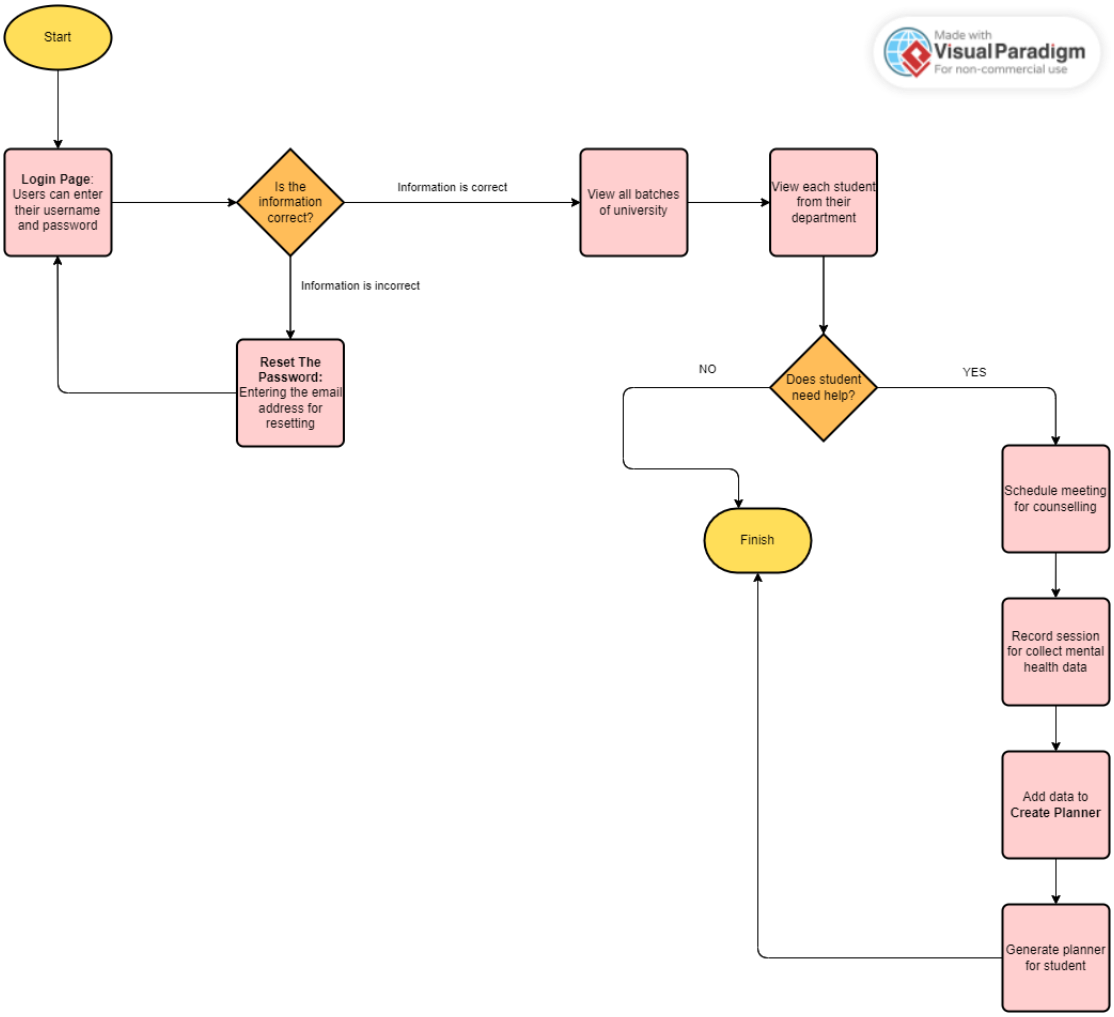


Figure 3 Process Flow Representation

CHAPTER 5: IMPLEMENTATION

5.1.1. *Algorithms*

Genetic Algorithm:

A genetic algorithm is a search and optimization technique inspired by the process of natural selection. It involves iteratively evolving a population of potential solutions to a problem by applying genetic operators such as selection, crossover, and mutation. The fittest individuals are more likely to pass on their traits to the next generation, leading to the discovery of optimal or near-optimal solutions over time. The five phases of a genetic algorithm are as follows:

1. Initial Population: In this phase, a population of potential solutions (individuals) to the problem is generated randomly or using some predefined criteria. Each individual is represented as a set of parameters or genes.

2. Fitness Function: The fitness function evaluates how well each individual in the population solves the problem at hand. It assigns a fitness value to each individual, which reflects its quality as a solution. The fitness function guides the algorithm by providing a quantitative measure of how close an individual is to an optimal solution.

3. Selection: The selection phase involves choosing individuals from the current population to act as parents for the next generation. Individuals with higher fitness values have a higher chance of being selected, simulating the principle of "survival of the fittest." This process increases the likelihood of passing beneficial traits to the next generation.

4. Crossover: Crossover, also known as recombination, is a genetic operator that combines the genetic material of two parents to create one or more offspring. This process mimics the way genetic information is exchanged during sexual reproduction. By mixing and matching genes from parents, crossover helps explore new areas of the solution space and potentially produce better solutions.

5. Mutation: Mutation is another genetic operator that introduces small random changes to an individual's genetic makeup. It adds diversity to the population and prevents the algorithm from getting stuck in local optima. While crossover

focuses on combining existing genetic information, mutation introduces novel variations that might lead to improved solutions.

We used this algorithm to create a student academic planner, taking into account constraints such as the credit hour limit and whether the student has passed the prerequisite courses.

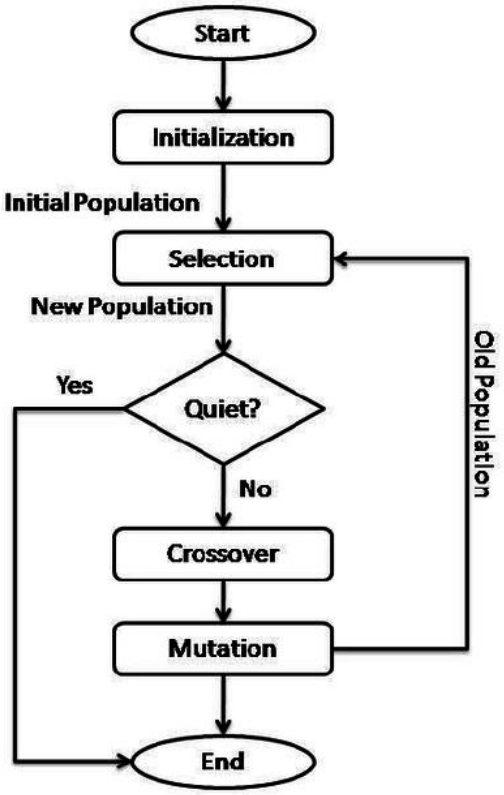


Figure 4 Genetic Algorithm Flow-Diagram

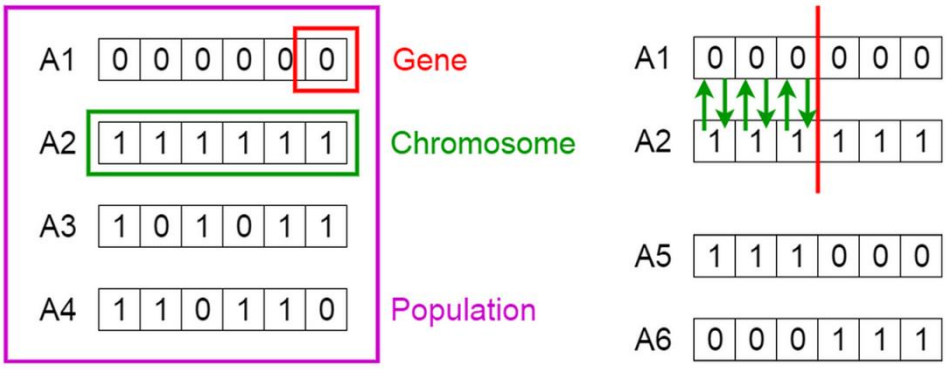


Figure 5 Visual Representation of Genetic Algorithm

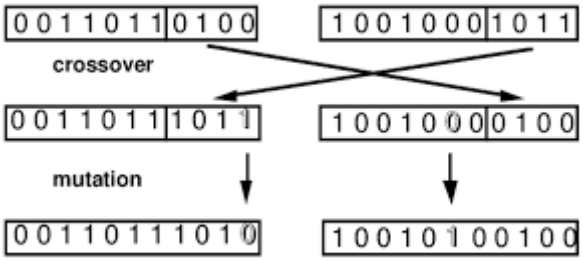


Figure 6 Crossover and Mutation

VGG16:

VGG16, short for the Visual Geometry Group 16, is a widely used convolutional neural network (CNN) architecture renowned for its effectiveness in image classification tasks. Comprising a total of 16 layers, VGG16 is characterized by its uniform 3x3 filter size across both convolutional and pooling layers, enabling the network to intricately learn patterns and features from images. This consistent architecture facilitates hierarchical feature extraction, making it particularly adept at discerning intricate details in visual data. In our specific application, VGG16 played a pivotal role in our emotion recognition model, where its deep structure and pattern recognition capabilities were harnessed to analyze the emotional states of students. By scrutinizing facial expressions, body language subtleties, and contextual cues, the model became proficient in deciphering various emotions. This integration of VGG16 substantially elevated the accuracy of our emotion predictions, as the model adeptly captured the nuanced visual elements indicative of distinct emotional states. This not only showcased VGG16's po-

tency in complex image analysis but also highlighted its significance in advancing computer vision applications like emotion recognition. We used VGG16 in our emotion recognition model to analyze the students' emotional state, leveraging its ability to learn complex patterns and features from images, thereby enhancing the accuracy of our predictions. The emotions we worked to predict were:

1. Happiness
2. Sadness
3. Fear
4. Disgust
5. Anger
6. Interest

By targeting these range of emotions, we aimed to create a comprehensive understanding of students' emotional states. This allowed our model to not only discern basic emotional categories like happiness and sadness but also capture nuanced feelings such as fear, disgust, anger, and interest. By considering this diverse spectrum of emotions, our model could provide a richer and more accurate analysis of students' emotional responses, enabling us to better support their emotional well-being and academic experiences.

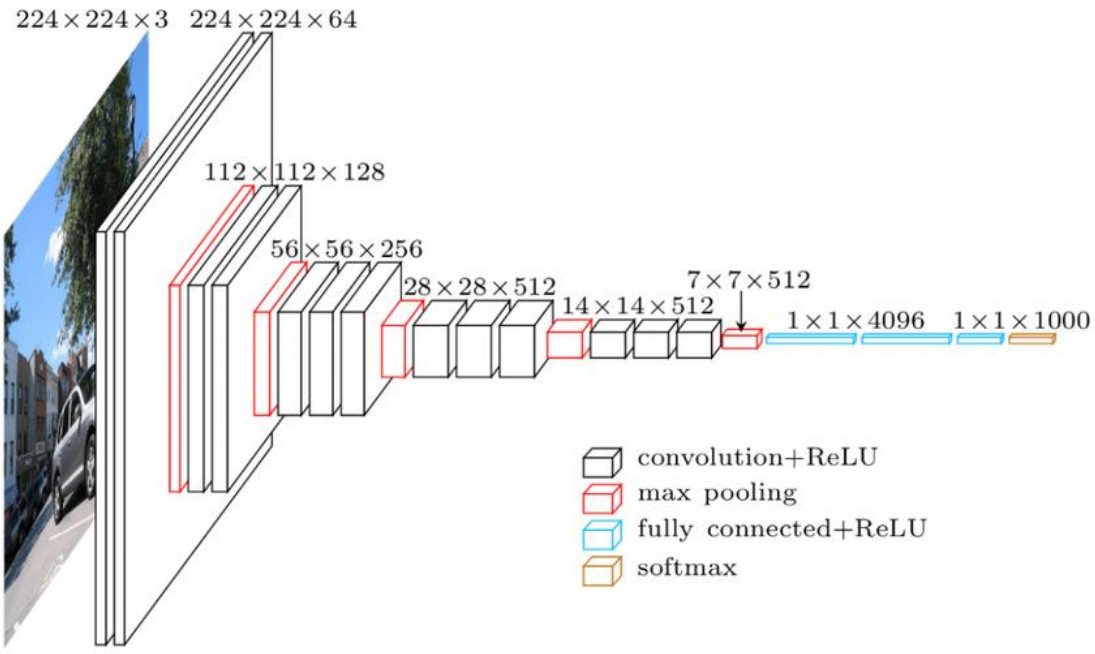
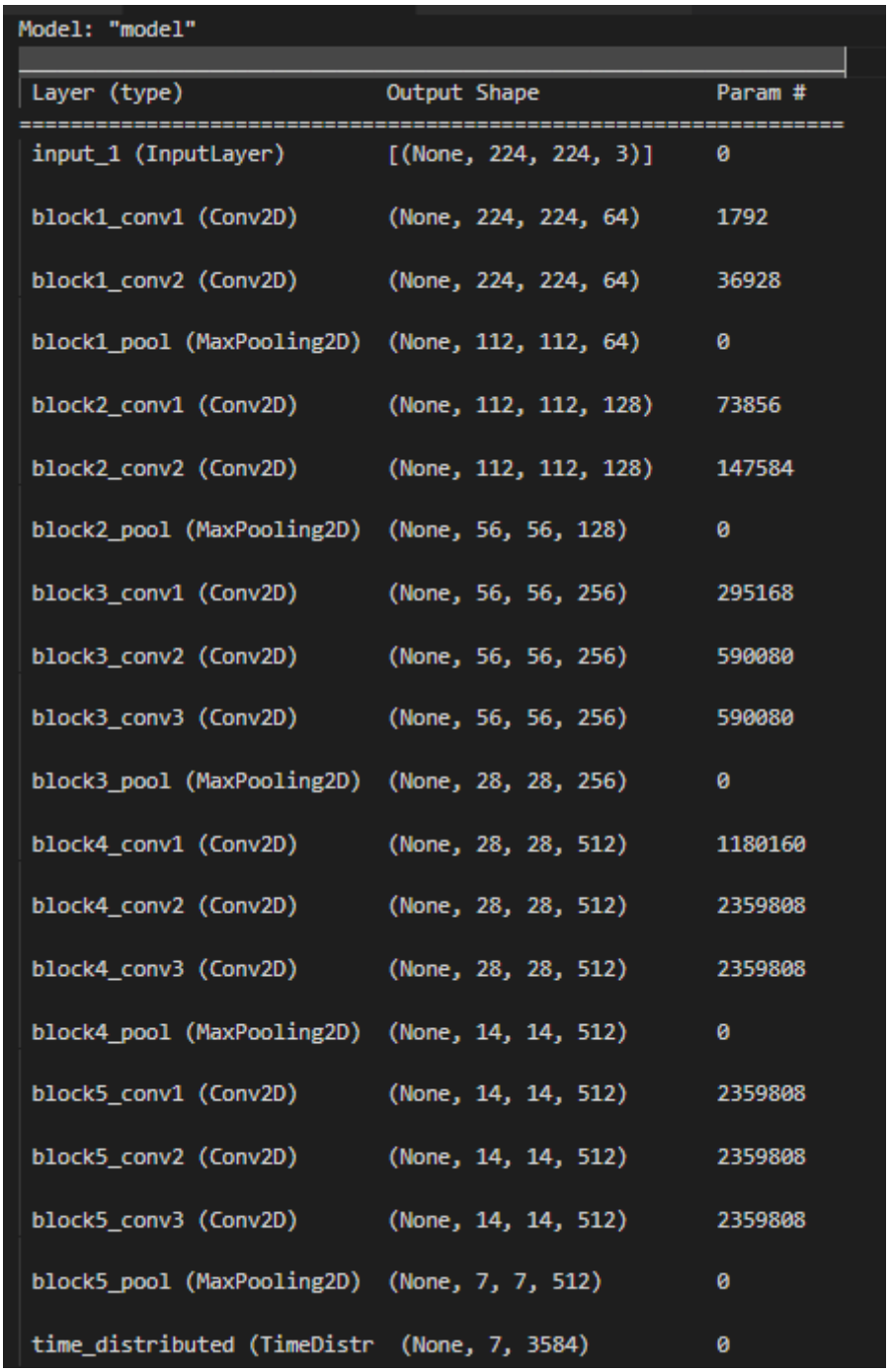


Figure 7 VGG16 Architecture Diagram

Following below fig. 8 and 9 shows the summary of the edited VGG16 model with it input and parameters it contains.



```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0

block1_conv1 (Conv2D)        (None, 224, 224, 64)      1792

block1_conv2 (Conv2D)        (None, 224, 224, 64)      36928

block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0

block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856

block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584

block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0

block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168

block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080

block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080

block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0

block4_conv1 (Conv2D)        (None, 28, 28, 512)       1180160

block4_conv2 (Conv2D)        (None, 28, 28, 512)       2359808

block4_conv3 (Conv2D)        (None, 28, 28, 512)       2359808

block4_pool (MaxPooling2D)   (None, 14, 14, 512)       0

block5_conv1 (Conv2D)        (None, 14, 14, 512)       2359808

block5_conv2 (Conv2D)        (None, 14, 14, 512)       2359808

block5_conv3 (Conv2D)        (None, 14, 14, 512)       2359808

block5_pool (MaxPooling2D)   (None, 7, 7, 512)         0

time_distributed (TimeDistr  (None, 7, 3584)           0
```

Figure 8 VGG16 Model Summary Part (a)

```
lstm (LSTM)                  (None, 256)               3933184

dense (Dense)                (None, 256)               65792

dropout (Dropout)            (None, 256)               0

dense_1 (Dense)              (None, 6)                 1542

=================================================================
Total params: 18,715,206
Trainable params: 4,000,518
Non-trainable params: 14,714,688
_________________________________________________________________
```

Figure 9 VGG16 Model Summary Part (b)

```
Epoch 1/15
745/745 [==============================] - 4470s 6s/step - loss: 0.9769 - accuracy: 0.6384
Epoch 2/15
745/745 [==============================] - 4559s 6s/step - loss: 0.5100 - accuracy: 0.8198
Epoch 3/15
745/745 [==============================] - 4832s 6s/step - loss: 0.3809 - accuracy: 0.8675
Epoch 4/15
745/745 [==============================] - 4944s 7s/step - loss: 0.3186 - accuracy: 0.8886
Epoch 5/15
745/745 [==============================] - 4757s 6s/step - loss: 0.3078 - accuracy: 0.8938
Epoch 6/15
745/745 [==============================] - 4609s 6s/step - loss: 0.2231 - accuracy: 0.9233
Epoch 7/15
745/745 [==============================] - 4620s 6s/step - loss: 0.2132 - accuracy: 0.9240
Epoch 8/15
745/745 [==============================] - 4571s 6s/step - loss: 0.1886 - accuracy: 0.9365
Epoch 9/15
745/745 [==============================] - 4561s 6s/step - loss: 0.1736 - accuracy: 0.9411
Epoch 10/15
745/745 [==============================] - 4568s 6s/step - loss: 0.1701 - accuracy: 0.9412
Epoch 11/15
745/745 [==============================] - 4567s 6s/step - loss: 0.1414 - accuracy: 0.9524
Epoch 12/15
745/745 [==============================] - 4637s 6s/step - loss: 0.1376 - accuracy: 0.9527
Epoch 13/15
...
Epoch 14/15
745/745 [==============================] - 4576s 6s/step - loss: 0.1394 - accuracy: 0.9526
Epoch 15/15
745/745 [==============================] - 4545s 6s/step - loss: 0.1200 - accuracy: 0.9589
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

Figure 10 VGG16 Model Accuracy

This model was trained on a laptop which had the following specifications;

- **CPU** Intel i5-8250u
- **RAM** 16 GB
- **iGPU** Intel UHD 620

It took close to two days to train the model on the following model.

5.1.2. External Features

Firebase:

Firebase stands as a robust cloud-based platform furnished by Google, playing a pivotal role in facilitating the storage and administration of data for a diverse array of applications. This comprehensive platform offers an assortment of essential services, including real-time database management, authentication mechanisms, hosting capabilities, and the execution of cloud functions. Through the seamless integration of Firebase into our application's architecture, a multitude of advantages are harnessed. All data generated and utilized within the application is securely stored and effortlessly accessed through Firebase's cloud infrastructure. The hallmark feature of real-time synchronization ensures that any updates or changes to the data are instantly propagated across all connected devices or instances, fostering a consistently up-to-date user experience. Moreover, Firebase's cloud-based nature grants the application an inherent scalability, meaning that as the demands and user base expand, the platform effortlessly accommodates the increased load without compromising performance. This scalability is an invaluable asset, offering the assurance that the application can grow in tandem with user needs. In essence, by incorporating Firebase into our application's foundation, we empower it with a reliable and efficient data management ecosystem that not only ensures data security and accessibility but also contributes to a seamless, real-time, and scalable user experience across a wide spectrum of potential applications.

Flask:

Flask serves as a lightweight and versatile web framework specifically designed for the Python programming language, and its seamless integration into the architecture of the student academic planner application significantly enhances its functionality. This framework acts as the structural backbone of the application, providing a systematic approach to organizing and managing various components. By utilizing Flask, the development process is streamlined, enabling the swift creation and incorporation of an

array of features and application programming interfaces (APIs) through the power of Python code.

Flask's lightweight nature doesn't compromise its capabilities; instead, it ensures efficiency and agility throughout the development lifecycle. The framework's modular design and extensive library support empower developers to construct intricate functionalities without excessive overhead. This modularity fosters a modular design philosophy where individual components can be developed, tested, and integrated with remarkable ease.

The integration of Flask within the student academic planner application particularly simplifies the development of dynamic features and interactive components. Python's clean syntax is combined with Flask's user-friendly routing system, which guides how URLs are mapped to corresponding functions. This empowers developers to create endpoints that handle various requests, such as retrieving data, updating information, or rendering dynamic content, all in a logical and organized manner.

Moreover, Flask inherently encourages the adherence to the Model-View-Controller (MVC) architectural pattern, which promotes a separation of concerns and a more organized codebase. This separation allows developers to work on different aspects of the application independently, enhancing collaboration and reducing the risk of conflicts during development.

In essence, the integration of Flask within the student academic planner application brings about an enriched development experience, combining Python's flexibility with Flask's structured approach. This pairing not only expedites the creation of diverse features and APIs but also lays the foundation for a coherent, well-organized, and maintainable application architecture that can be extended and adapted to suit the evolving needs of the user community.

5.1.3. Home Page

The application starts with eye pleasing animation. This animation represents the logo of our application.

Figure 11 Loading Page

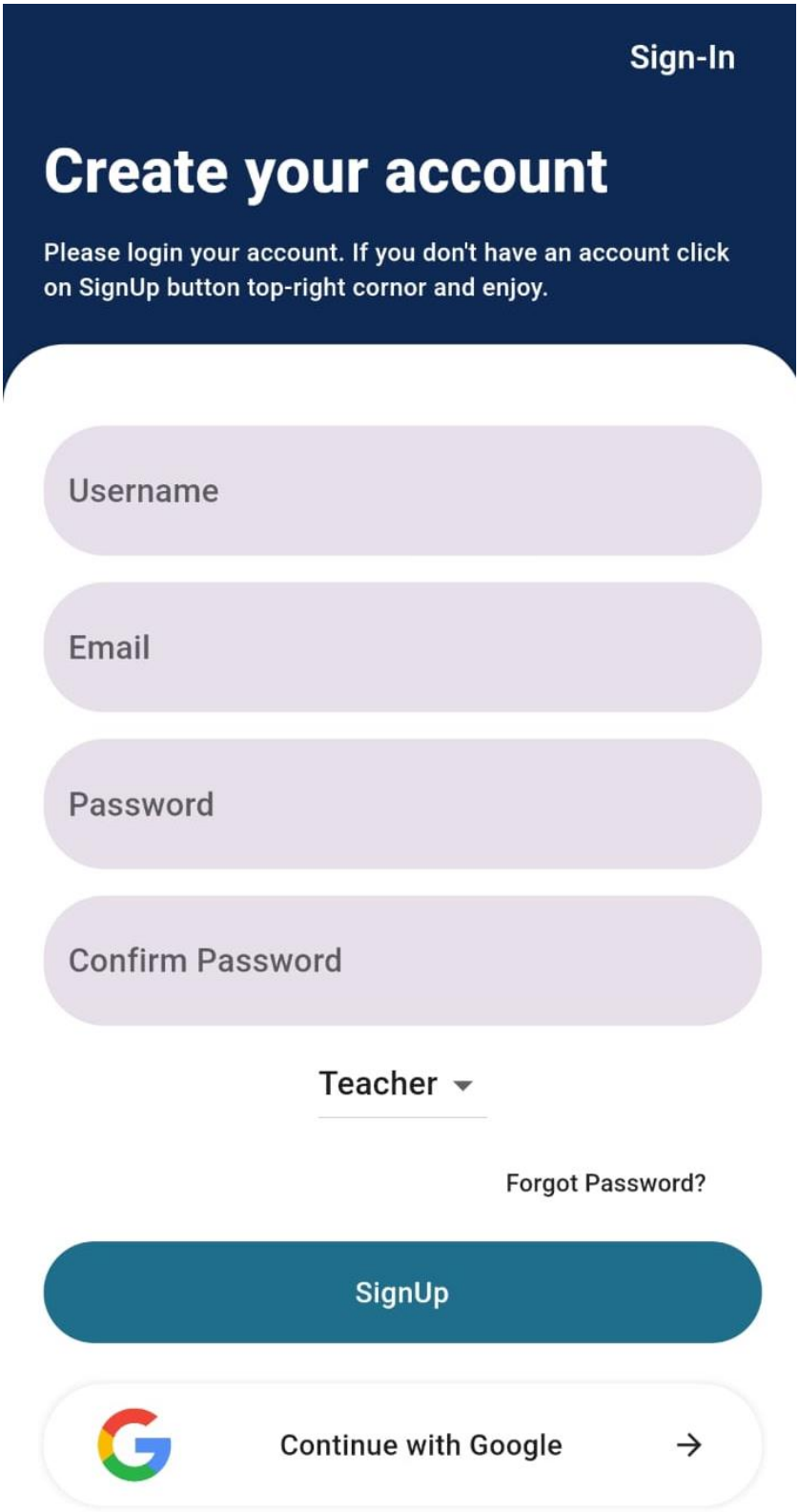


Figure 12 Sign Up Page

The signup page marks the initial step for individuals to join the student academic planner application, offering a seamless and secure route to create personalized accounts based on their roles within the system. Within this user-friendly interface, three distinct

options are presented: students and teachers, each designed to cater to specific user categories and roles within the educational ecosystem.

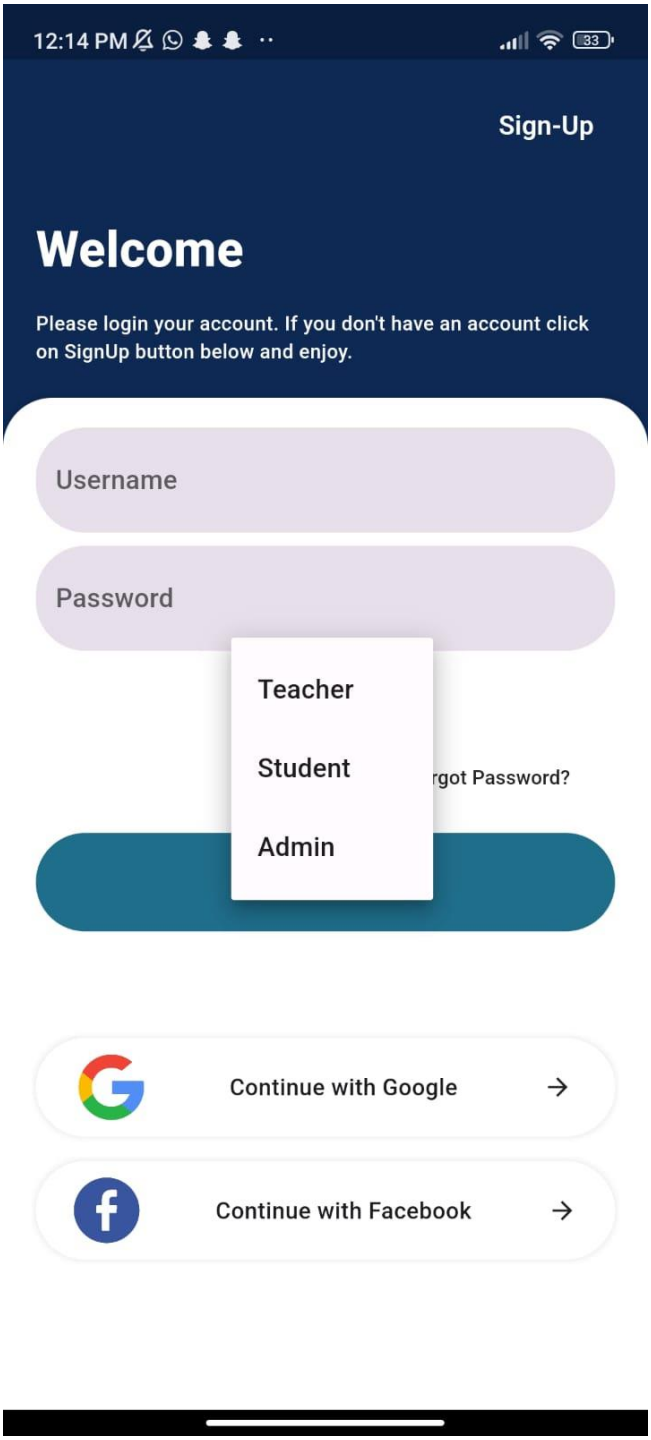


Figure 13 Login Page

The login page represents the gateway to the student academic planner application, offering users a seamless and secure entry point to their respective roles within the system.

This intuitive interface presents three distinct options: students, admin, and teachers, each tailored to cater to specific user categories.

5.1.4. *Admin Interface*

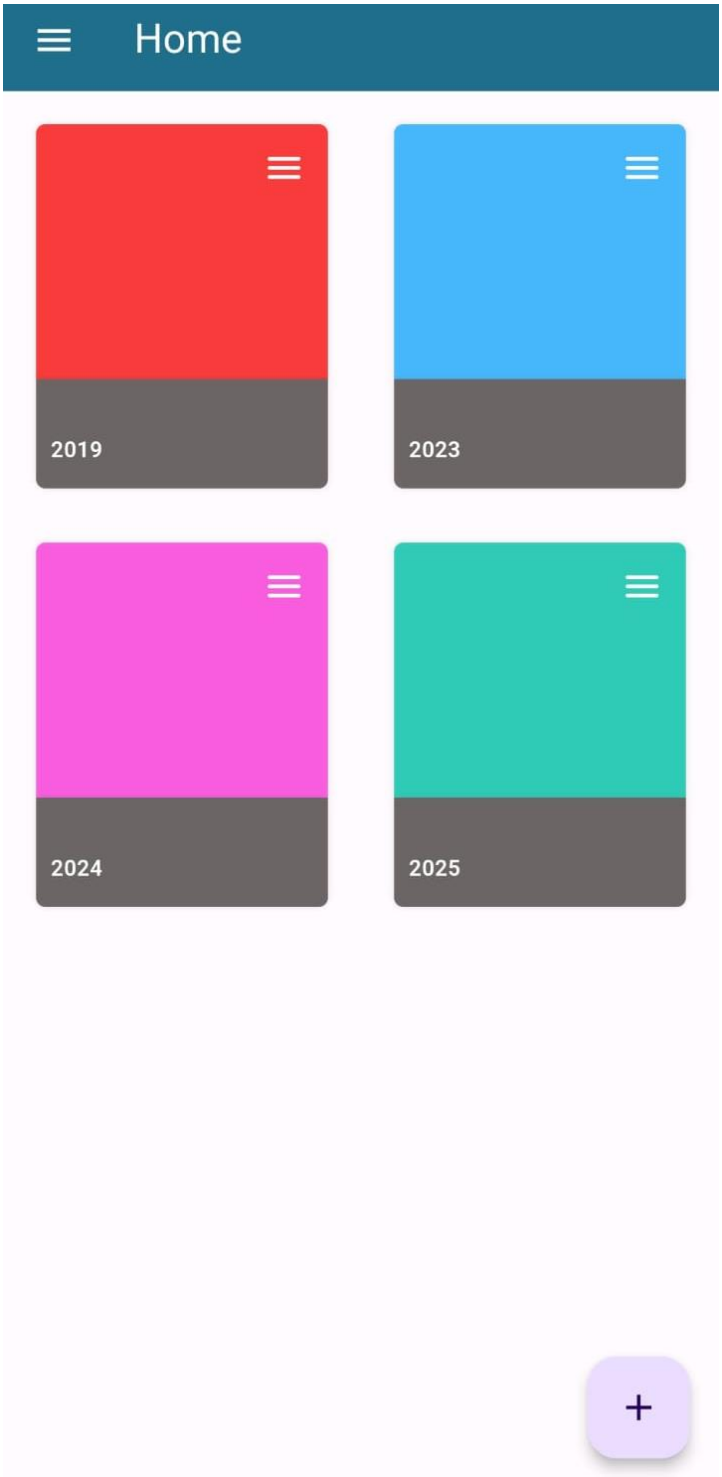


Figure 14 Dashboard to Access Batches

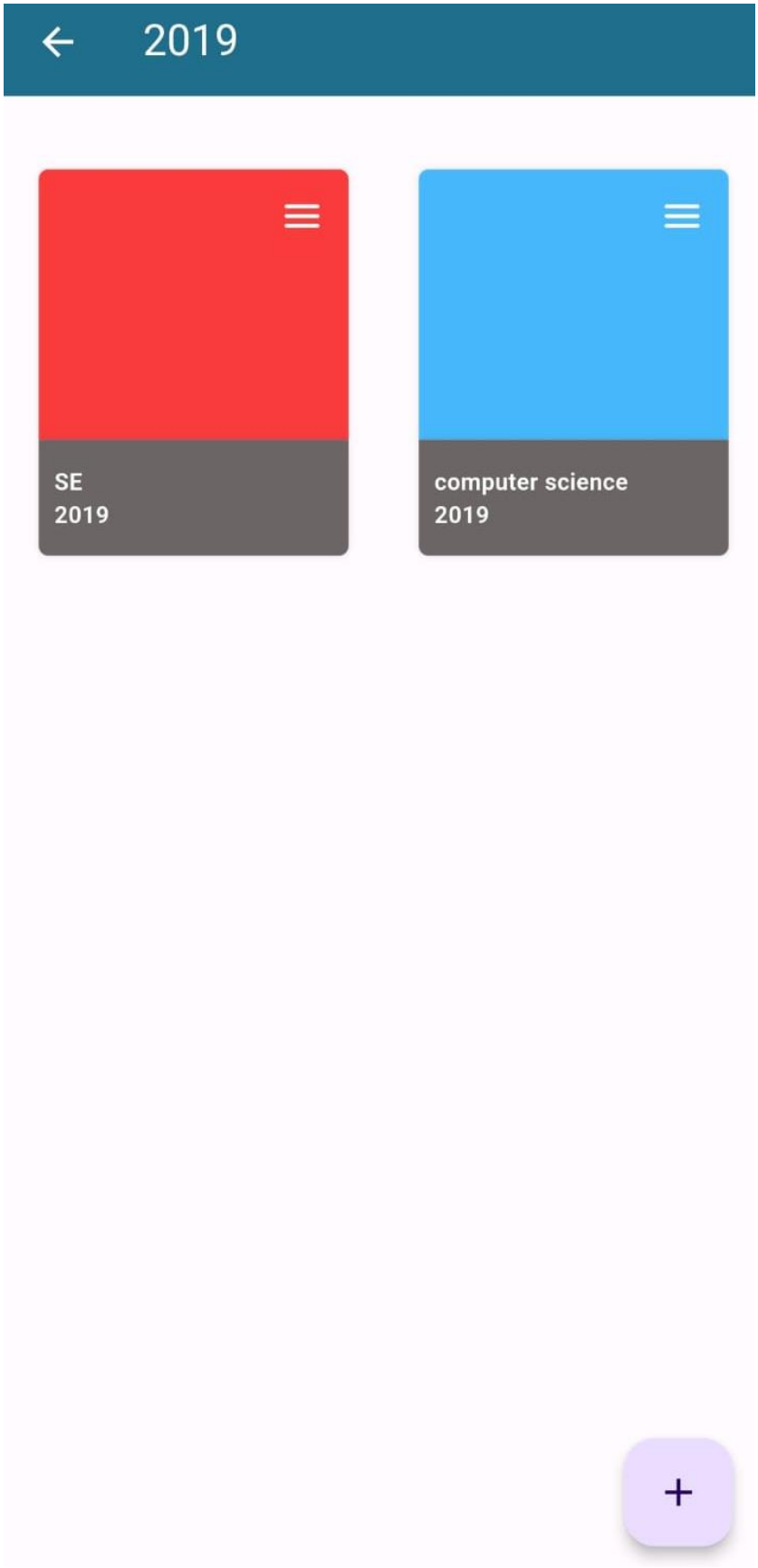


Figure 15 Dashboard to Access each Department in a Batch

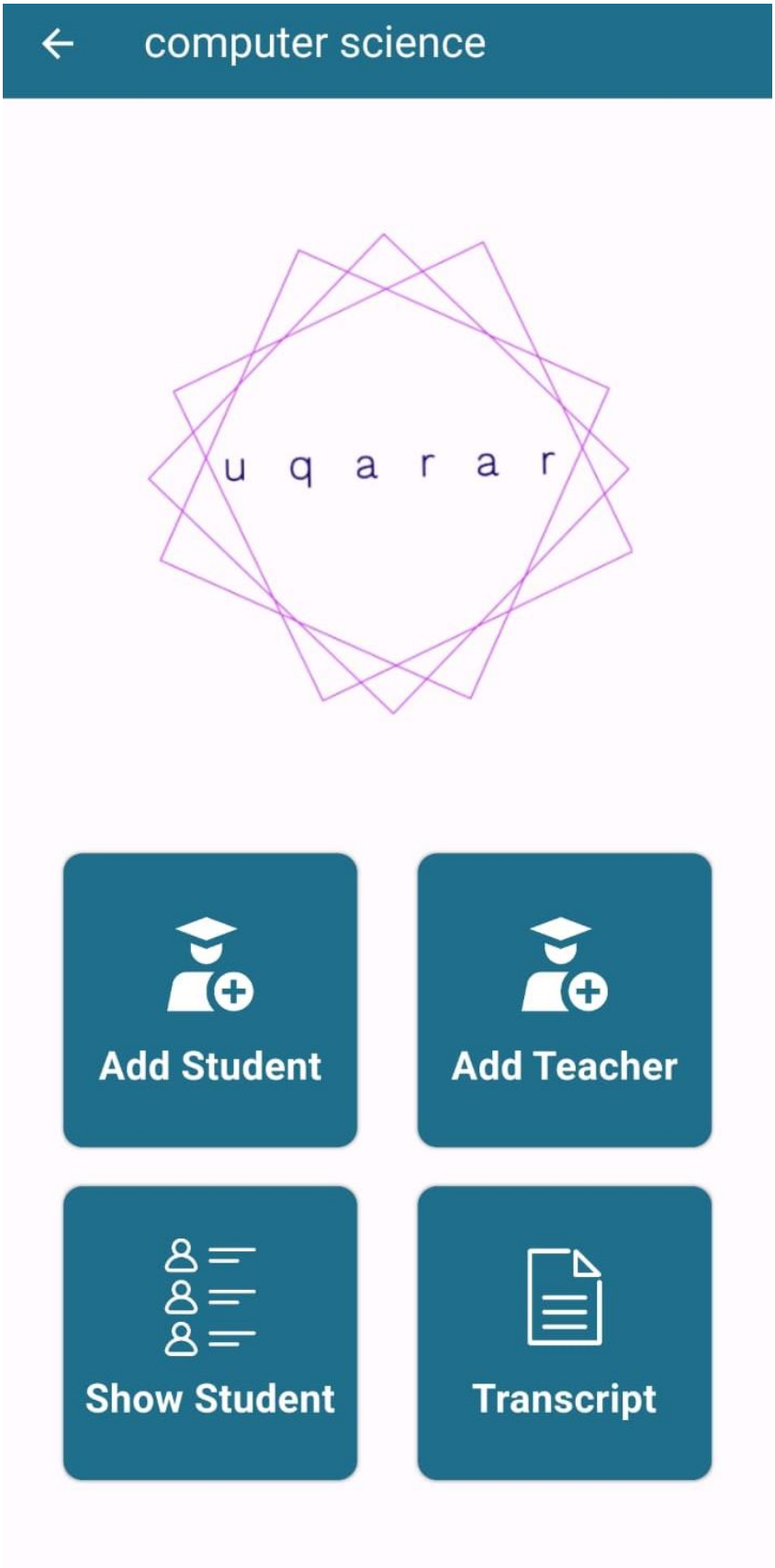


Figure 16 Control Panel for Admin

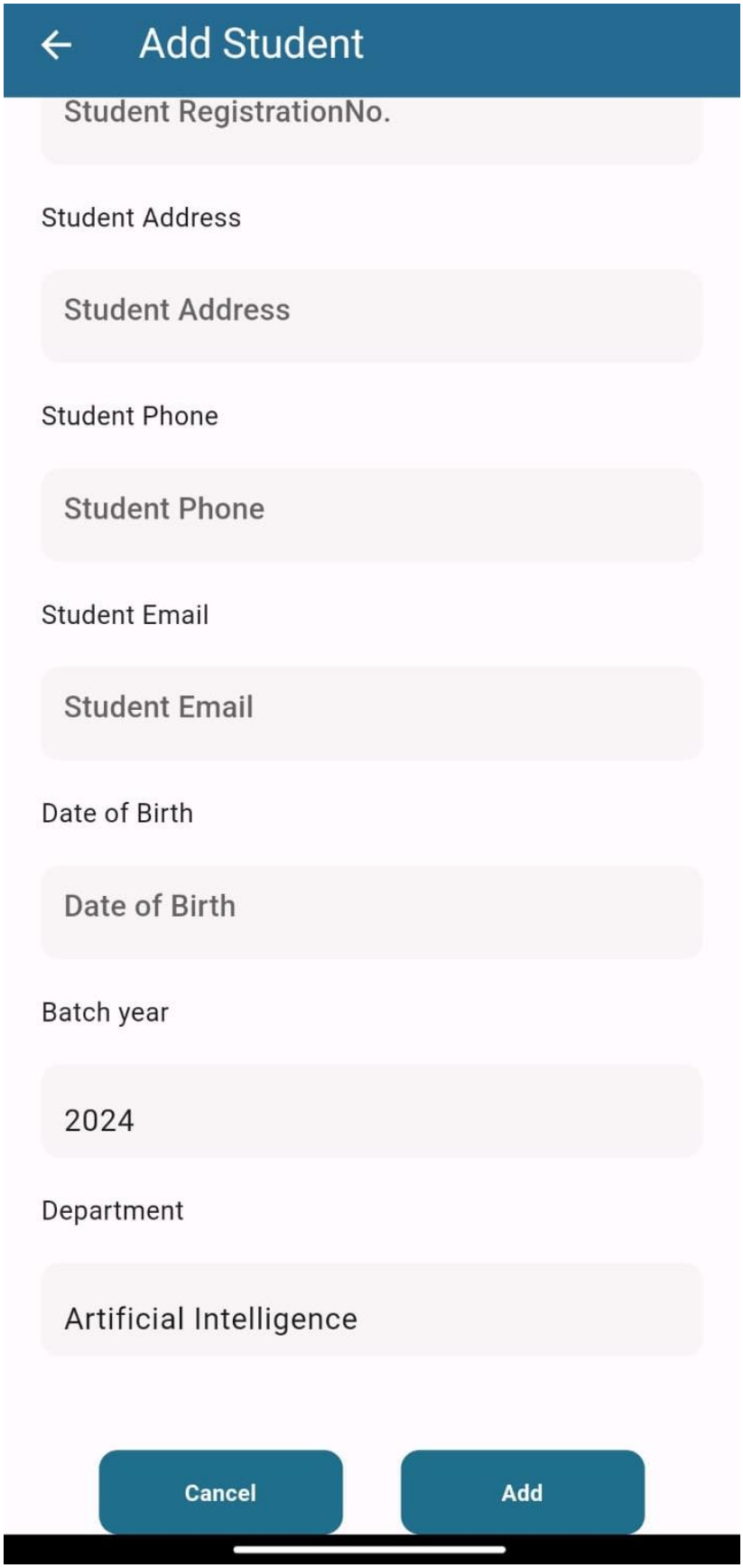


Figure 17 Student Information

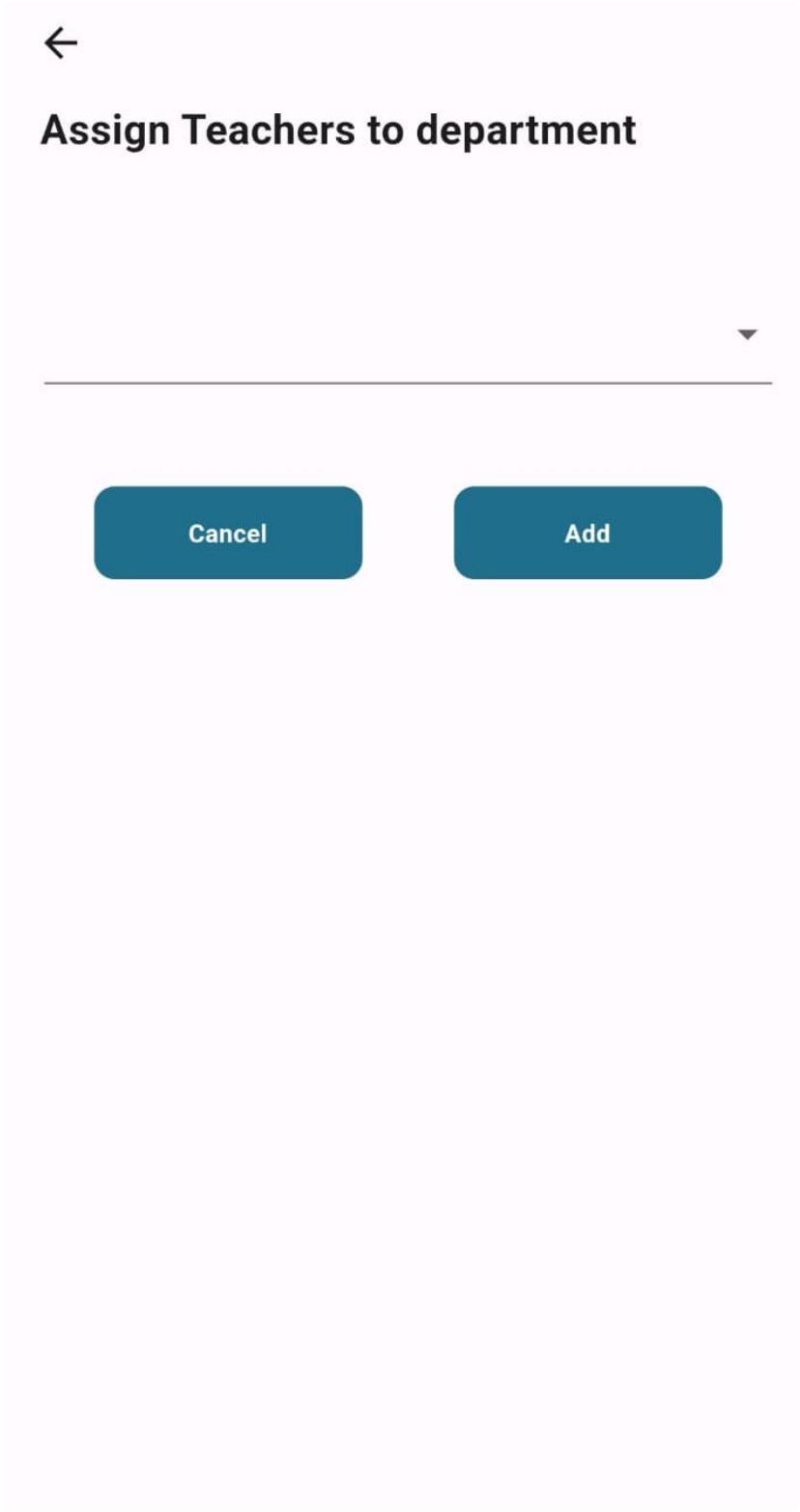


Figure 18 Assigning Teachers to Department

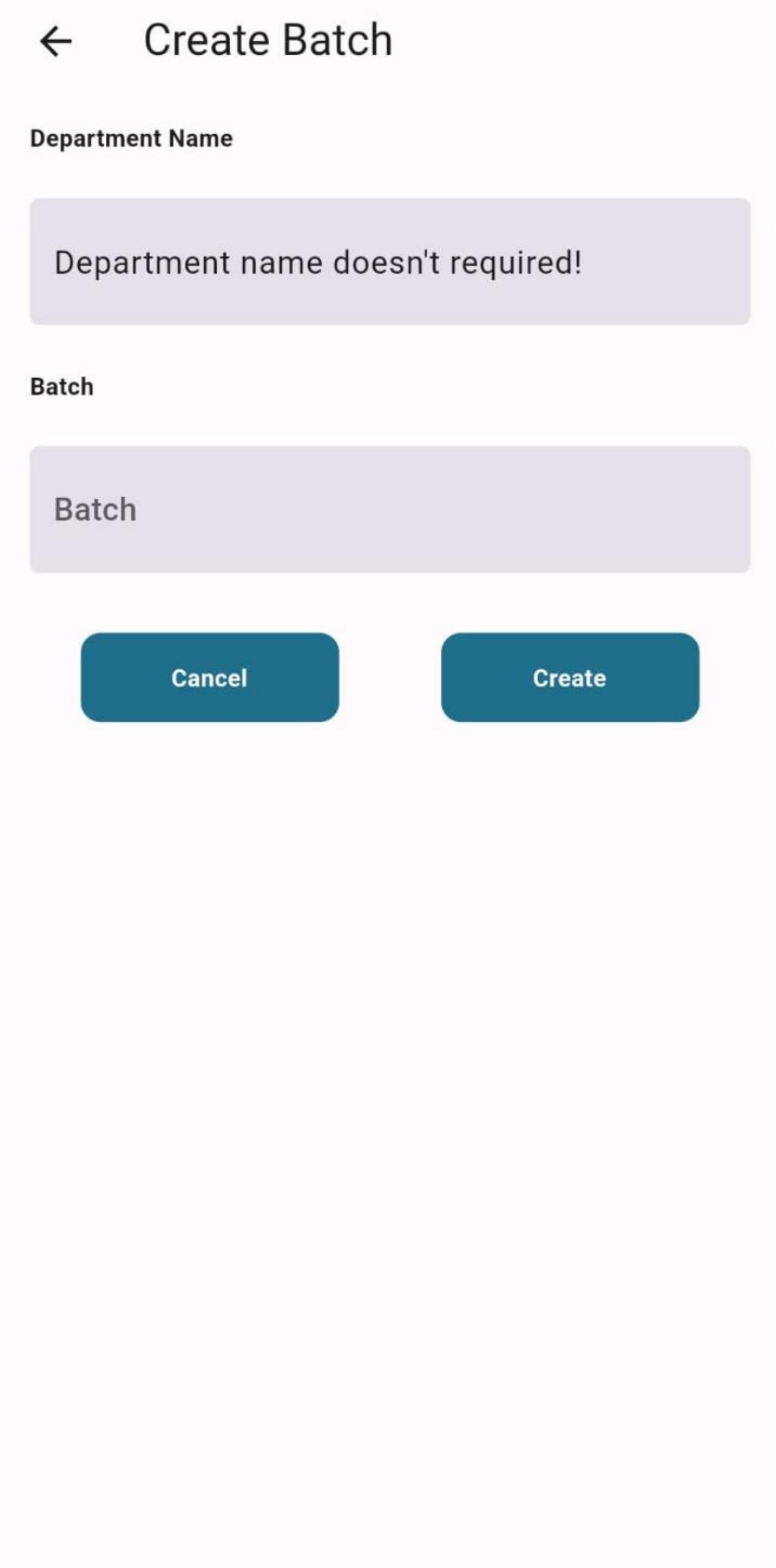


Figure 19 Create Batch

5.1.5. Teacher Interface

Figure 20 Teacher Dashboard

Figure 21 Departments Assigned to Teacher

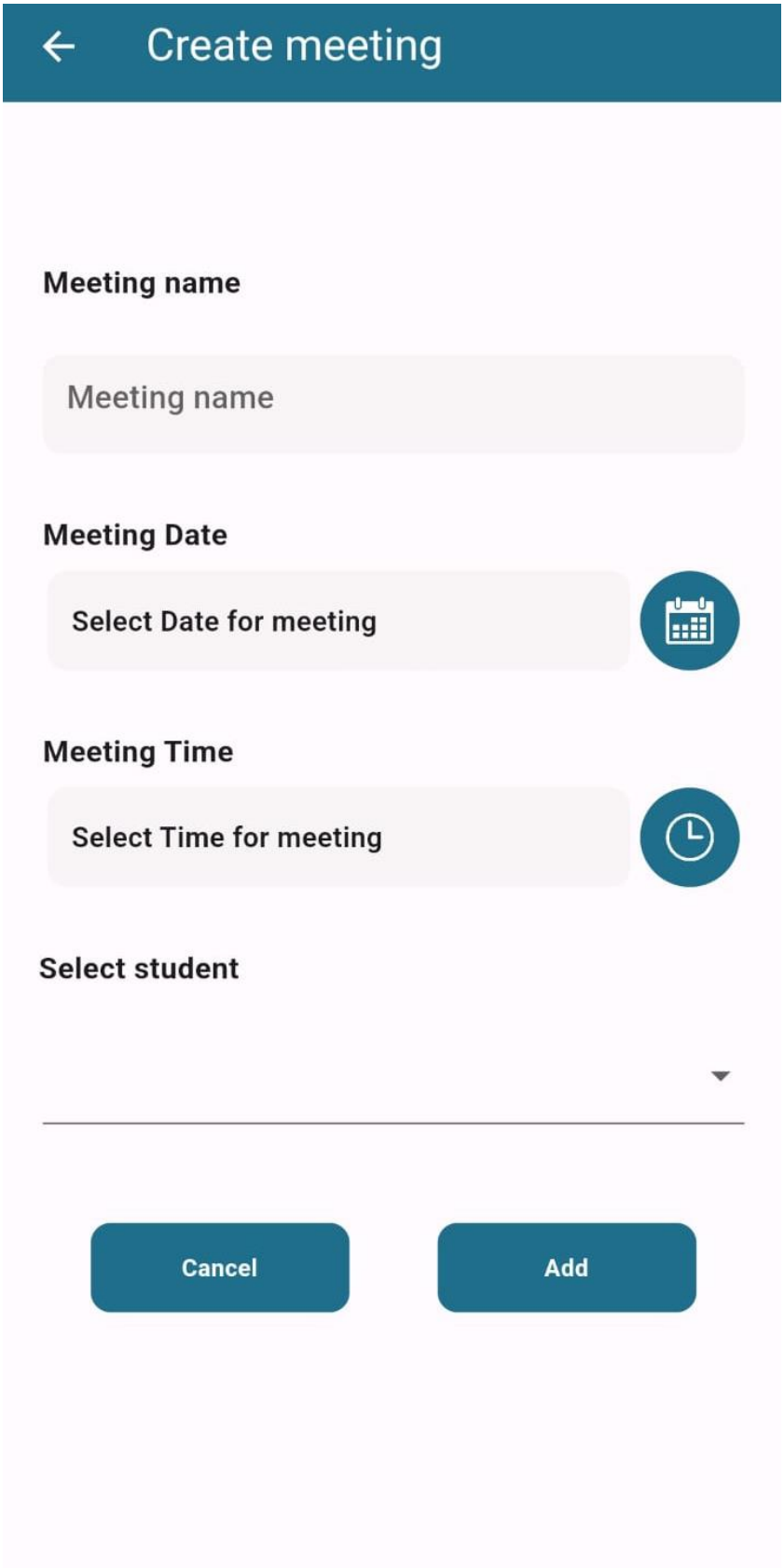


Figure 22 Create Meetings

Figure 23 Meeting Dashboard

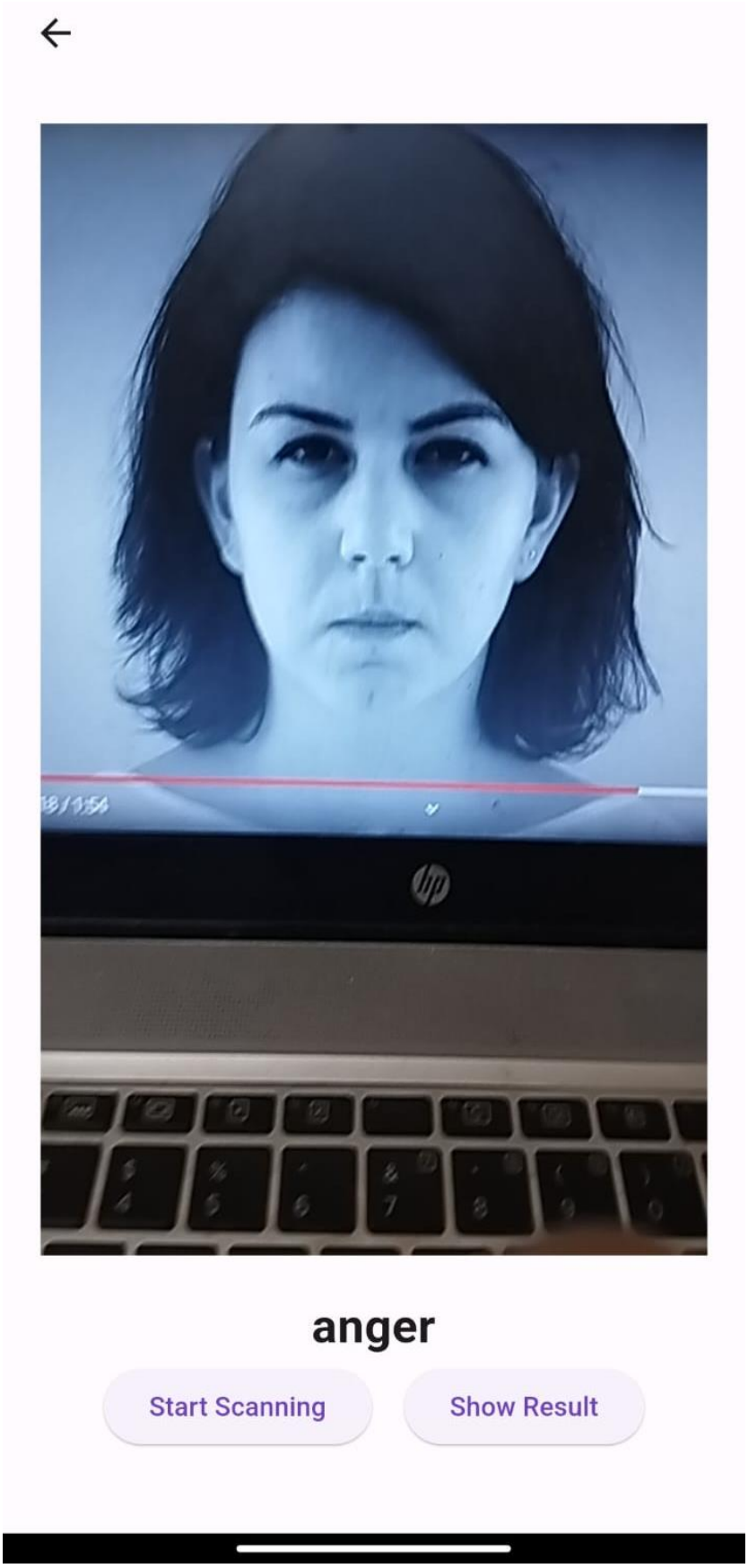
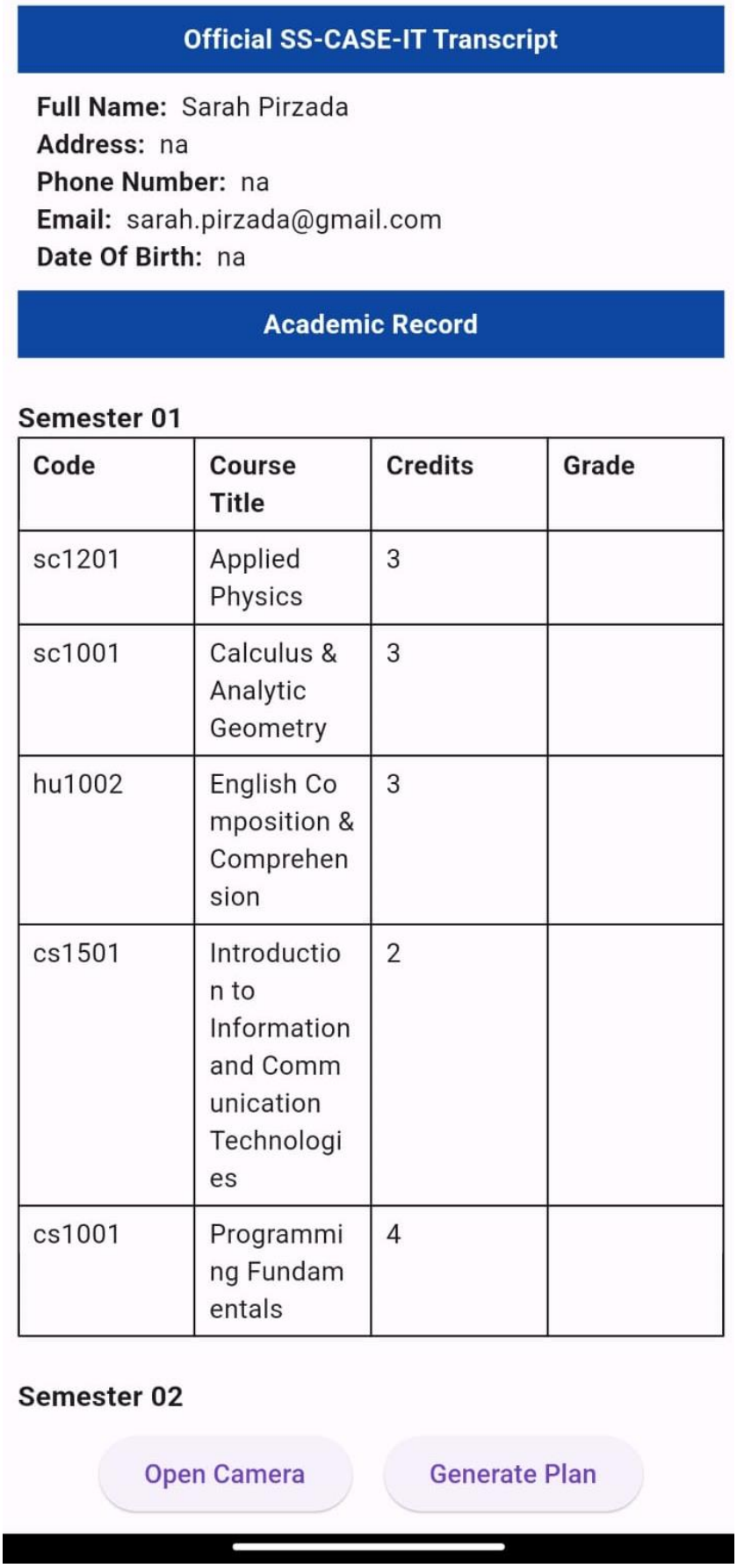


Figure 24 Emotion Recognition Camera

Figure 25 Student List



Official SS-CASE-IT Transcript

Full Name: Sarah Pirzada
Address: na
Phone Number: na
Email: sarah.pirzada@gmail.com
Date Of Birth: na

Academic Record

Semester 01

Code	Course Title	Credits	Grade
sc1201	Applied Physics	3	
sc1001	Calculus & Analytic Geometry	3	
hu1002	English Composition & Comprehension	3	
cs1501	Introduction to Information and Communication Technologies	2	
cs1001	Programming Fundamentals	4	

Semester 02

Open Camera

Generate Plan

Figure 26 Student Transcript

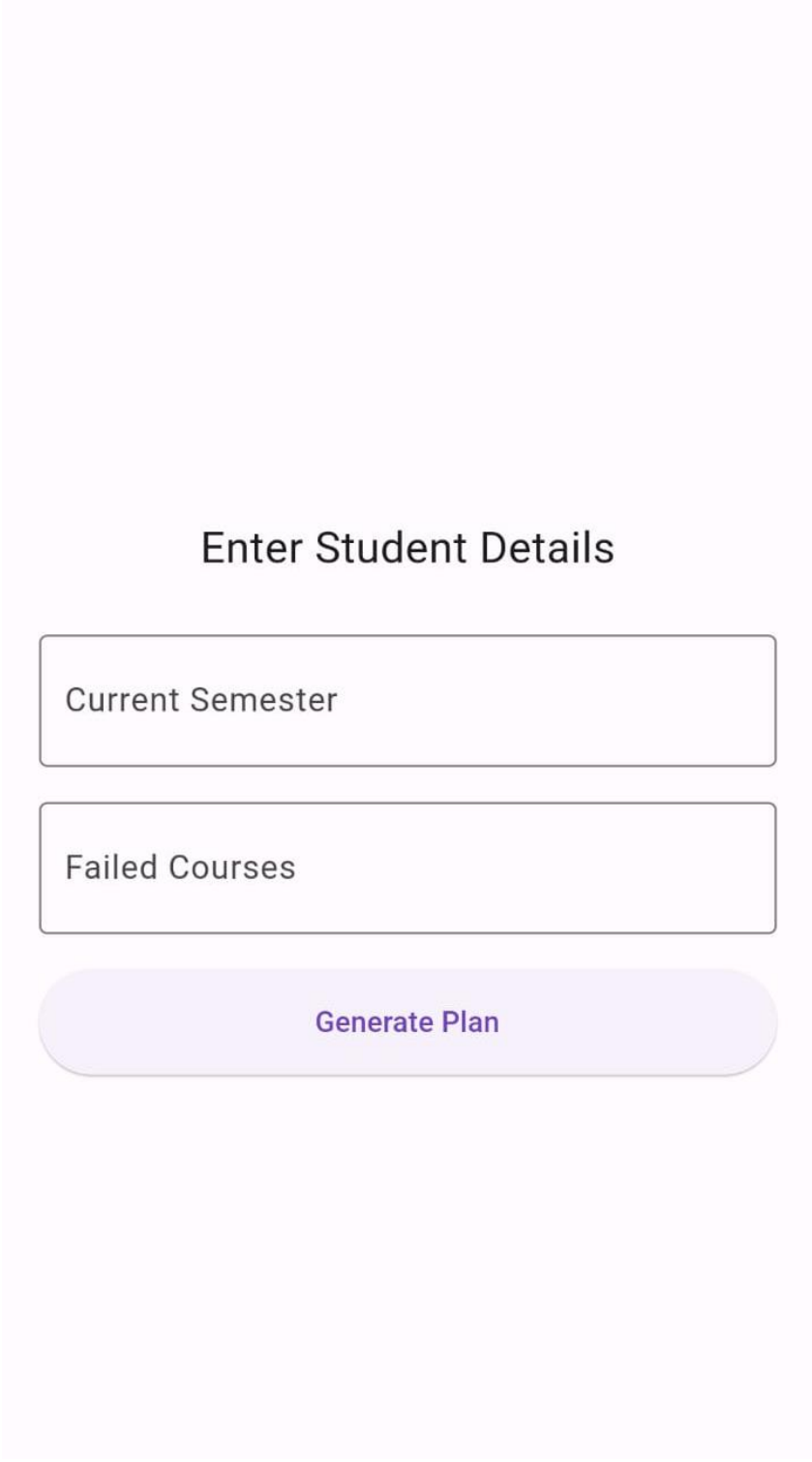


Figure 27 Planner Generator

5.1.6. Student Interface

Figure 28 Dashboard for Students to Check their Meetings

CHAPTER 6: TESTING AND EVALUATION

6.1.1. *Testing Methodology*

The testing methodology devised for our mobile application system, with its overarching goal of creating optimal academic progression plans for students that incorporate their mental health considerations, comprises a meticulously crafted sequence of pivotal stages. At its core, this methodology is designed to establish a robust and reliable application. The process commences with functional testing, a meticulous examination undertaken to ascertain the flawless operation of each feature and functionality. This thorough scrutiny takes into account various course conditions and prerequisites, ensuring that the application aptly adapts to diverse scenarios. Concurrently, usability testing takes center stage, aimed at gauging the user experience. To achieve this, the insights of both undergraduate students and student counselors are garnered, cultivating an understanding of their perceptions and interactions. Their valuable feedback, channeled into iterative refinements, shapes an interface that aligns seamlessly with user needs.

The methodology further extends to encompass performance testing, a pivotal evaluation that validates the application's responsiveness and stability under diverse conditions. This phase scrutinizes how the application fares under varying loads and interactions, ensuring that it delivers a consistent and smooth experience even in demanding scenarios. Integral to this holistic testing framework is integration testing, a crucial measure that ensures the harmonious synchronization between our application and the genetic algorithm it employs. This intricate partnership is pivotal in generating the finely-tuned academic progression plans that cater not only to educational needs but also to mental well-being considerations.

Collectively, these testing stages coalesce into a comprehensive approach that meticulously examines different dimensions of the application's functionality, usability, performance, and overall acceptance. This multifaceted approach engenders a level of confidence in our system's capabilities, ensuring that it not only performs as intended but also seamlessly integrates mental health considerations into the fabric of academic planning. As a result, our application emerges not just as a technological tool, but as a holistic solution striving to optimize academic paths while nurturing students' well-being.

CHAPTER 7: CONCLUSION AND FUTURE WORK

7.1.1. *Conclusion*

In embarking on this project, our initial aim was to develop a tool that would simplify the process for student counselors in creating academic plans for students with complex situations. Throughout the project, we had the opportunity to delve into new technologies, languages, and algorithms. As we progressed, we conducted extensive testing and gathered insights from both our own observations and valuable feedback from counselors within our faculty. This enabled us to identify and incorporate the essential features necessary to enhance the functionality and usefulness of the tool.

Beyond the technical aspects, this project was a journey of growth and learning. We encountered challenges that tested our problem-solving skills and ingenuity, while also deepening our understanding of the intersection between technology and education. The collaboration with faculty members and counselors brought real-world perspectives to our work, aligning it more closely with the needs of the academic community.

Our commitment to addressing the complexities of academic planning for students experiencing unique challenges led to the development of a tool that not only streamlines the process but also fosters a greater sense of inclusivity and support within our educational environment. As we conclude this project, we reflect on the evolution of our goals, from a technical tool to a catalyst for positive change in academic counseling and student well-being.

7.1.2. Future Work

Our vision is rooted in an ongoing commitment to elevate and broaden the horizons of this project, forging a trajectory that encompasses both growth and enrichment. At its core, our aspiration revolves around the cultivation of a dynamic ecosystem that not only adapts to change but actively seeks to amplify its impact.

Our journey forward is guided by several key principles and aspirations. First and foremost, we are dedicated to continually infusing the project with additional features, carefully chosen to expand its capabilities. These features are thoughtfully curated to address a spectrum of needs and concerns, further enhancing the application's utility in the academic planning process.

Central to our journey is the unwavering dedication to the continuous enhancement of the application's functionality. Through meticulous refinement and strategic augmentation, we aim to evolve the project into a multifaceted tool that encapsulates a comprehensive suite of features. This evolution is not limited to functionality alone; we also place great importance on the aesthetic dimension. An intuitive and engaging interface is paramount, as it is the conduit through which users navigate and interact. By blending practicality with aesthetics, we aim to create an environment that is both utilitarian and inviting.

However, our aspirations extend beyond technological advancement. We envision this project as a catalyst for tangible, positive change in the educational landscape. Every enhancement, feature addition, and design refinement contribute to a tool that fundamentally enriches academic planning and, by extension, student success.

Our overarching goal is to extend the reach of this initiative across a broader spectrum of educational institutions. By doing so, we aim to maximize its influence and reach, ensuring that our work has a lasting and transformative impact on the academic community.

In essence, our journey is underscored by a deep-seated commitment to making education more accessible, efficient, and effective for all involved, and we remain dedicated to the continuous improvement and expansion of our project to achieve these objectives.

REFERENCES

[1] Srisamutr, T. Raruaysong and V. Mettanant

"A Course Planning Application for Undergraduate Students Using Genetic Algorithm," 2018 Seventh ICT International Student Project Conference (ICT-ISPC), Nakhonpathom, Thailand, 2018, pp. 1-5, doi: 10.1109/ICT-ISPC.2018.8523980.

[2] S. Dwijayanti, M. Iqbal and B. Y. Suprapto,

"Real-Time Implementation of Face Recognition and Emotion Recognition in a Humanoid Robot Using a Convolutional Neural Network," in IEEE Access, vol. 10, pp. 89876-89886, 2022, doi: 10.1109/ACCESS.2022.3200762.

[3] VGG 16 Guide

Everything you need to know about VGG16 | by Great Learning | Medium

[4] Genetic Algorithm Guide

Genetic Algorithm with Python by Clinton Sheppard