School of Computer Sciences

CAT404 – Software Engineering Major Project

System Requirement and Design Document

[Student Personal Assistance]

[No Group 16: Fresas]

Student Name	Matric Number	Subsystem Name	Code	Role
Muazah Binti Kamal Bahari	147029	My Study Life (iStudy)	SE22230040	Team Leader/ Requirements Analyst
Simrenpreet Kaur a/p Charanjit Singh	147158	My Social Network (MyCircle)	SE22230041	Quality Assurance Analyst
Nur Syahirah Binti Ibaharim	149214	Fitness Track and Healthy Eating (beFit)	SE22230042	Architect

Supervisor: Dr Noor Farizah Ibrahim

Examiner 1: Assoc. Prof. Mohd Azam Osman

Examiner 2: Dr. Mohd Adib Haji Omar

Academic Session

2022/2023

Declaration

"We declare that the following is our own work and does not contain any ***unacknowledged*** work from any other sources. This report was undertaken to fulfill the requirements of the Software Engineering Major Project for the Bachelor of Computer Science (Honours) program at Universiti Sains Malaysia".

Signature : 

Name : Muazah Binti Kamal Bahari

Date : 7 January 2023

Signature : 

Name : Simrenpreet Kaur a/p Charanjit Singh

Date : 7 January 2023

Signature : 

Name : Nur Syahirah Binti Ibaharim

Date : 7 January 2023

Abstrak

Student Personal Assistance adalah aplikasi mudah alih (telefon pintar) yang menawarkan bantuan kepada pelajar khususnya untuk keperluan pelajar Universiti Sains Malaysia. Student Personal Assistance adalah platform yang dapat memudahkan penjadualan pelajar mengikut keselesaan mereka, bila-bila masa dan di mana sahaja. Aplikasi ini dibina berdasarkan tiga subsistem - My Study Life (iStudy), My Social Network (MyCircle), dan Fitness Track and Healthy Eating (beFit). Ia adalah aplikasi mudah alih yang dapat menangani masalah yang dihadapi oleh pelajar dari segi pengurusan masa, kemahiran sosial, dan pengekalan gaya hidup sihat. Ia menyatukan tiga subsistem, setiap dengan tujuan, fungsi, dan ciri yang unik, memberikan pelajar dengan satu penyelesaian yang komprehensif untuk keperluan mereka. Untuk projek ini, Metodologi Agile akan digunakan semasa membina aplikasi mudah alih ini. Kajian digunakan dalam proses pengumpulan maklumat untuk tujuan pengumpulan maklumat tambahan yang diperlukan mengenai keperluan sistem. Setelah selesai projek ini, pelajar akan mempunyai alat dan sistem yang diperlukan untuk mengimbangi dan menguruskan tanggungjawab akademik, sosial, dan keperluan kesihatan personal dengan berkesan.

Kata Kunci: iStudy, MyCircle, befit, Metodologi Agile

Abstract

Student Personal Assistance is a mobile app that offers students personal assistance features that is specifically tailored to the needs of University Science Malaysia students. Student Personal Assistance is a platform that can facilitate the scheduling of students at their convenience, anytime and anywhere. This app is built based on three subsystems– My Study Life (iStudy), Social Network (MyCircle), and Fitness Track and Healthy Eating (beFit). It is a cutting-edge mobile app that addresses the key challenges of time management, social skills, and maintaining a healthy lifestyle faced by university students. It seamlessly integrates three subsystems, each with its own unique purpose, functions, and features, providing students with a single, comprehensive solution for their needs. For this project, Agile Methodology will be utilized when developing this mobile application. A survey is used for the requirement-gathering purpose to collect any additional information needed about the requirements of the system. Upon completion of this project, students will have the tools and resources they need to effectively balance and manage their academic, social, and personal health commitments.

Keywords: iStudy, MyCircle, beFit, Agile Methodology

Acknowledgements

We would like to express our most profound appreciation to several individuals and organizations for supporting us throughout this project (Milestone 1: System Requirement and Design Document). First, we wish to express our sincere gratitude to our former supervisor, Ts. Dr. Sharifah Mashitah Syed Mohamad and our current supervisor, Dr Noor Farizah Ibrahim, for their insightful comments, helpful information, practical advice and unceasing ideas that have helped us tremendously to organize our project Student Personal Assistance mobile app well and for the better development of our project.

Moving on, we would like to express our gratitude to our coordinator, Ts. Dr. Heikal for providing us with the greatest assistance and guidance in order to implement our report well and for all the information given throughout the journey of developing this project successfully. We also wish to express our appreciation to our examiner panels, Assoc. Prof. Mohd Azam Osman and Dr. Mohd Adib Haji Omar. Finally, we would like to greatly thank our family and friends for their constant support and encouragement throughout our studies and the time allocated for this project. The support given along provided us strength to continue our project development.

Table of Contents

List of Figures

List of Tables

1. Software Project Management Plan (SPMP)

1.1. Project background

At present, students have a hard time having stabilized or organized lives. They can plan their study life and follow the schedule for a short amount of time before getting busy and demotivated. Eventually, they will become neglectful and refuse to follow it. These will affect their academics, social life and health. This is where Student Personal Assistance comes to fruition.

Student Personal Assistance is highly related to Sustainable Development Goal 4 (SDG 4) as the connections between education and health are diverse and bi-directional; health is not only compromised of physical well-being but also mental and social. Having no stress in academic life can lead to the students having a healthy mental state and indirectly their social life.

Education is essential for sustainable development, but investments in education that are not accompanied by concurrent progress in other aspects of human well-being will fall short of enabling all people to realize their full potential in life [1].

This project is focused on developing a system that will ultimately be used as a personal assistant for students. We are motivated because we are students who must manage time, be organized, balance social and academic life, and stay fit and healthy. There are a few apps that can help students manage their schedules, social networks, and health and fitness, but they are all separate apps. As a result, it is difficult for students to balance their lives at university. Student Personal Assistance combines all of these subsystems into a single system that assists students in managing their academic, social, and fitness lives. This mobile application is intended to make it easier for students to use and practise by integrating three separate subsystems with distinct goals, functions, and features into a single app.

Education allows societies to lay the groundwork for future success by passing on knowledge, values, and skills from generation to generation. Hence, Student Personal Assitance can be a tool in ensuring education will strive for this purpose

due to its closely related to SDG 4 is to ensure inclusive and equitable quality education and promote lifelong learning opportunities for all that starts in university [1].

1.2. Organization background

Student Personal Assistance is owned by University Science Malaysia (USM). The system's users will be university students in USM. They will use the system by downloading the app through their mobile phones as the system developed is a mobile application. Student Personal Assistance would benefit the organization in terms that it would help the students to balance their academics, social, and health. These 3 are the most important elements for a student that is vital to be taken care of. Having a single application that helps to manage those and have access to them will indeed help the student to a great degree of convenience. Without proper management of this, it results in the students are constantly stressed, unorganized, unhealthy, and lonely. Furthermore, it also helps the students to be more productive as it efficiently manages their life and it improves the general performance of the students which would indirectly impact positively to the organization as well.

1.3. System Overview

1.3.1. System Description and Function

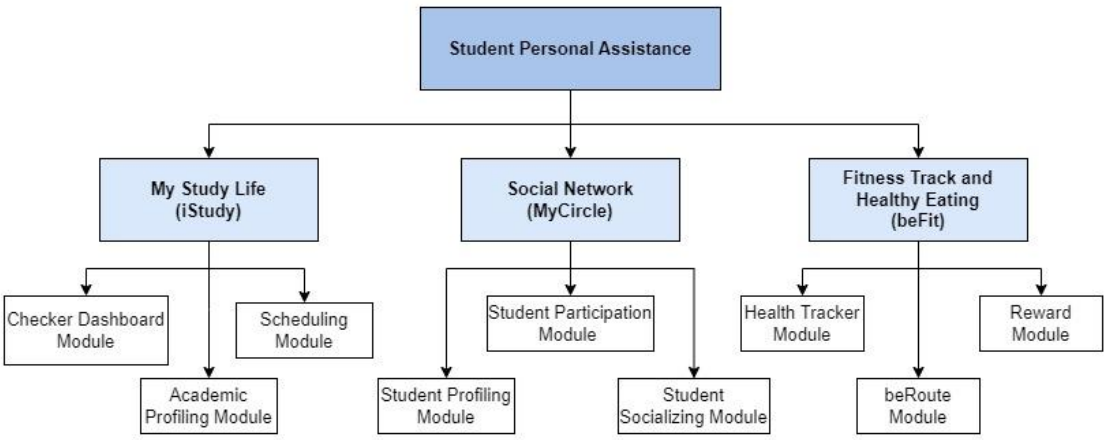


Figure 1.1: System Overview

Student Personal Assistance is tailored to meet the needs of university students, specifically those at USM. This application is designed and developed to make it

easier for university students to manage their personal responsibilities, and to help them become organized and efficient, leading to increased productivity and improved performance. For students, a mobile application that helps them to manage their academic, social and healthy lives is essential. It is crucial for students to have access to such a tool to help them navigate their daily lives. Student Personal Assistance can be a valuable resource for students who want to achieve their academic goals, improve their time management, and develop social skills while maintaining a healthy lifestyle. Acquiring these skills and habits during their university years can greatly reduce the stress of student life.

Student Personal Assistance consists of 3 main subsystems with each of them having 3 modules. The first subsystem is called iStudy which behaves as a time-management tool that assists in organizing university students´ schedules such as timetables and tasks. Viewing the system´s landing page (dashboard) as well as allowing students to view their academic, social and health performance. Students can view academic profiling that allows the student to utilize many functionalities. The second subsystem, called MyCircle, functions as a social platform for students to connect with each other and participate in clubs and activities on campus. One notable feature of MyCircle is the recommender system, which suggests student social clubs based on the student's interest. It also includes a chat feature to encourage engagement among students. Through MyCircle, students can earn participation marks for their academic records and benefit from the social connections and opportunities it offers. Lastly, the third subsystem called beFit is developed for health tracking in terms of fitness tracking and healthy eating, rewarding achievements after completing goals, and using a real-time route feature to display the route taken and calculate the steps and calories burned for the particular route intended to encourage students to adopt a healthy lifestyle. What differs between students having a healthy lifestyle and the public is students can develop complete self-awareness. University is an excellent setting for teaching young adults how to improve their healthy lifestyle behaviors, such as mental health, social skills, and academic performance. It is critical to assess young adults' health behaviors during this time period and to provide them with assistance where needed.

1.3.2. Software Process Approach

The software process approach that has been chosen to develop the Student Personal Assitance mobile application is Agile. The reason agile is chosen is that agile is an iterative approach of project delivery that emphasizes continuous releases that take client feedback into account. Student Personal Assistance is divided into sprints or smaller work and is manageable so it can be completed within a few weeks. Besides, agile is also ideal for projects with changing requirements. Planning workflows to anticipate changes and be adaptable enough to identify new solutions is a major agile idea [2]. This can be necessary to address unforeseen problems or if new development emerges. This makes the process to add or delete features as our project moves along easier.

Agile also helps us to identify problems or bugs early as the testing is done at the end of each sprint rather than at the completion of the project. This can ensure that defects or bugs are corrected as soon as possible. Additionally, the development process of agile frameworks enables us as individuals and teams to work independently on the project and complete tasks at our own pace and in our own way. Last but not least, agile allows for flexibility because we can never predict when something unforeseen will happen to our team or process thus agile makes it simple to reorganize our team and assign tasks in different priority order.

1.3.3. Software Life Cycle Model

The Software Life Cycle Model that is implemented in Student Personal Assistance is Scrum. Scrum framework allows the implementation of Agile development methodology. Scrum allows an iterative and incremental development process where this whole project would be broken down into several incremental phases [3]. Each deliverable in these incremental phases would result in a ready-to-use product for the customer consequently, feedback is received at each of these phases to produce a better product [3]. The small period of incremental phases mentioned is called sprints. These sprints tend to be about 1 to 4 weeks.

There are 3 key roles in Scrum that are the Product Owner, the Scrum Master, and the Development Team. In our project, the Product Owner is University Sains Malaysia

(USM) while our Scrum Master is our leader in this project and the Development Team consists of the remaining members.

The artifacts in the scrum which are the product backlog and sprint backlog are implemented as well. To avoid confusion among us while we are developing the project, high-level requirements are finalized from the end-user perspective which is called user stories [4]. This is created in a document known as product backlog which consists of all user stories.

These tasks after prioritization are then further divided into sprints [4]. It consists of user stories that will be completed during the current sprint. Each of us will select the user story we would like to develop and begin on it. Each of us has to make sure that we will be capable of finishing all these stories on time. The document created is called sprint backlog. We conduct our scrum meeting every 3 days consecutively. Frequent changes and discussions are held on WhatsApp and Google Meet as it is a more convenient platform that gives rapid responses. Formal discussions and documentation is held in Microsoft Teams where each progress is updated in a folder and it allows access for editing to the other members.

To track the progress of each of us, a task board is used. This is where it is a table that consists of labels of “To Do”, “In Progress”, “Testing”, and “Done”. Every time each of us is working on the development phase of any user story we would then write down our progress on the respective label.

Besides, scrum meetings are held. In these meetings, the main goal is to get complete information about the current status of the project and to make sure that all of us are on the same page having the same understanding. In these scrum meetings, each of us would tell the progress that we have made for the Sprint Goal, the task to be done next, and any complication or confusion that is faced by any of us. Then we would discuss it and find a proper solution to it. Scrum meetings are held every 3 days consecutively.

At the end of each sprint, a sprint review would be conducted. This is where the end product of each sprint would be reviewed and analyzed. This is to ensure any amendments to be made and plan for the next sprint. A sprint retrospective is also conducted where it aims to discuss the end result produced and decide ways on how to

improve the development process for the next sprint. All of us would discuss the process of their development and decide what could be done better for future iterations.

The final phase is when the end product of each sprint is evaluated with the acceptance criteria and demonstrated to the customer. Consequently, feedback is received from the customer which will then be utilized in the upcoming sprints. Values in Scrum such as committed, courage, focus, openness, and respect are implemented individually within us as it guides the behavior of the scrum team and helps in delivering value to the customer.

Product Backlog for Student Personal Assistance - User Stories.

- As a student, I want to be able to view my academic, social, and health performance in both table and chart form so that I can keep track of my performance during my university life.
- As a student, I want to be able to view course recommendations so that I can enroll in those recommended courses.
- As a student, I want to be able to keep track and manage my task and be allowed to set and receive reminders so that I would not forget the deadline.
- As a student, I want to be able to update my basic information and social interests so that my profile remains updated with the latest information.
- As a student, I want to be able to view social events held and participate in events that I am interested in so that I can gain participation marks from it.
- As a student, I want to be able to view a list of other students using this application, search for friends to add, and option to add them as friends so that I can expand my social circle at university.
- As a student, I want to be able to manage my food intake and exercise diary by inputting my daily food intake and exercises I have done so that it will be easier to keep track and manage my health in university.
- As a student, I want to be able to set reminders for mealtimes for breakfast, lunch, and dinner so that I would not skip my meals and be reminded when I am occupied with work until I forget to input my daily food intake.

- As a student, I want to be able to view my points after completing goals in the rewards so that I would cherish my hardworking journey and strive to keep adopting a healthy lifestyle.

1.3.4. Modeling Notation

Student Personal Assistance is developed by effectively applying Unified Modeling Language (UML). UML is a standardized modelling language consisting of a set of diagrams. It was created to assist system and software developers in specifying, visualizing, building, and documenting the components of software systems [5]. Each UML diagram has a specific goal or answers questions such as What does the Student Personal Assistance app do? Who's the Student Personal Assistance app interacting with? And how? What's the structure of the Student Personal Assistance source code?

UML has 2 types of diagrams; static and dynamic. Static diagrams describe unchanging relationships between different items. Dynamic diagrams describe how different items collaborate or communicate over time [5]. This project uses use case diagrams and class diagrams for static and sequence diagrams and activity diagrams for dynamic.

Use-case diagrams in UML model the behaviour of a system and aid in the capture of system requirements. Use-case diagrams describe a system's high-level functions and scope. These diagrams also show how the system and its actors interact with each other. In use-case diagrams, the use cases and actors describe what the system does and how the actors interact with it, but not how the system operates internally. Use-case diagrams depict and define the context and requirements of an entire system or key components of a system. Student Personal Assitance only has 1 actor which is the student that interacts with the system.

When engineers need to change the app, they must first understand how the code works. They must understand the main components and how they are related. This also allows for identifying the design patterns—or the lack thereof [5]. A class diagram is required when the items to be documented are classes. Class diagrams are to model the objects that make up the system, display the relationships between the objects, and describe what those objects do and the services that they provide [6]. As class diagrams

expand and grow, they may become incoherent. It's best to avoid creating large diagrams and instead divide them into smaller ones that can be linked together later. This is why developing a simple and straightforward in our project is critical to improving readability.

During the analysis and design phases of the development cycle in Student Personal Assistance, class diagrams perform the following functions; capture and define the structure of classes and other classifiers, define relationships between classes and classifiers, depict the structure of a model by using attributes, operations, and signals, display the common classifier roles and responsibilities that define the behaviour of the system, and display the structure and behaviour of one or more classes [6].

Sequence Diagrams capture the interaction that occurs in a collaboration that either reflects a use case or an operation, as well as high-level interactions between system users and the system, the system, and other systems, or subsystems. Instead of cluttering our sequence diagrams with several objects and groups of messages that will hinder readability, a few smaller sequence diagrams corresponds to our use cases in the use case diagrams that aptly explain what each of our subsystems does.

The activity diagram, which is similar to a flowchart or a data flow diagram, visually represents a series of actions or the flow of control in our system. In business process modelling, activity diagrams are frequently used. They can also use a use case diagram to describe the steps. Modelled activities can be both sequential and concurrent [7].

1.3.5. Coding Standard

Coding standards differ from person to person and can be different for different projects, especially for Student Personal Assistance which has three subsystems with a different focus that has to be integrated into one system. It is extremely crucial to have coding standards that ensure all developers working on the project adhere to the project's specifications. The codes also need to be simple to understand, and proper consistency is maintained. The code can be easily understood and proper consistency is maintained. Consistency has a positive impact on program quality and should be maintained while coding. It should also be ensured that coding rules are followed consistently across all levels of the system and do not contradict one another. The

finished program code should appear to have been written in a single session by a single developer.

This project is developed by using Android Studio which is the Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. The coding standard that is most important for developers to adhere to is being consistent. Developers should spend a few minutes determining the coding style and sticking to it until the end. It should be the same if that code uses spaces around the if clauses. If the code comments are surrounded by tiny boxes with stars, then the comments should be similarly surrounded.

There are rules to follow when using Android's Java libraries and tools. In some cases, the convention has shifted significantly, and older code may employ a deprecated pattern or library [8]. It's fine to stick with the current style when working with such code. However, developers should never use deprecated libraries when creating new components. When possible, developers should keep methods short and focused. They must recognise that long methods are sometimes necessary, so no hard limit on method length is imposed. If a method has more than 40 lines, consider whether it can be broken up without affecting the program's structure.

Keep the scope of local variables as small as possible. This improves the readability and maintainability of the project's code while decreasing the possibility of error. Developers must declare each variable in the innermost block that contains all of the variable's uses. They must also declare local variables at the point where they are used for the first time. An initializer should be included in almost every local variable declaration [8].

Developers must be cautious of security leaks through logs, so avoiding logging private information is critical. Avoid logging information about protected content in particular. This is especially important when writing framework code because it is difficult to predict what will and will not be private information or protected content.

1.4. Team Structure and Roles

1.4.1. Role Assignments

There are several roles that have been assigned to each team member. First and foremost, each team member is automatically a developer, and one must be a Team Leader. The following additional roles that are assigned are Requirements Analyst, Architect, and Quality Assurance Analyst.

Role	Team Member
Project Leader / Requirement Analyst	Muazah Binti Kamal Bahari
Quality Assurance Analyst	Simrenpreet Kaur a/p Charanjit Singh
Architect	Nur Syahirah Binti Ibaharim

Table 1.1: Project Team Role Assignments

1.4.2. Development Responsibilities

The following team members have been assigned to the given subsystems for the project.

Subsystem	Team Member
My Study Life (iStudy)	Muazah Binti Kamal Bahari
My Social Network(MyCircle)	Simrenpreet Kaur a/p Charanjit Singh
Fitness Track and Healthy Eating (beFit)	Nur Syahirah Binti Ibaharim

Table 1.2: Subsystem Development Responsibilities

1.5. Facilities and Computer Resources

1.5.1. Computer and other Hardware Resources

Minimum hardware requirements needed in the development of this system are:

Processor	- Intel Core i5 - AMD A6
Memory	8GB RAM
Storage	256GB SSD

Table 1.3: Hardware Equipment Requirements for System Development

1.5.2. Software and Operating System Resource Specifications

Software and operating system requirements needed and used to develop this system are:

Operating System	Google Android Operating System (OS)
Development Tools	• Android Studio • Java • Java JDK • IntelliJ IDE • Intel x86 Emulator Accelerator
Documentation Support Tools	• Microsoft Teams • Google Docs • Diagrams.net • Canva
Collaborative Software Development	• Git • GitHub

Table 1.4: Software and Operating Systems Resource Specifications for System Development

1.6. Project Schedule & Milestones

A phased project schedule is used for project development. There are three important milestones:

Milestone 1: System Requirement and Design

Milestone 2: Progress Review (Prototype development)

Milestone 3: Final Presentation (Final development)

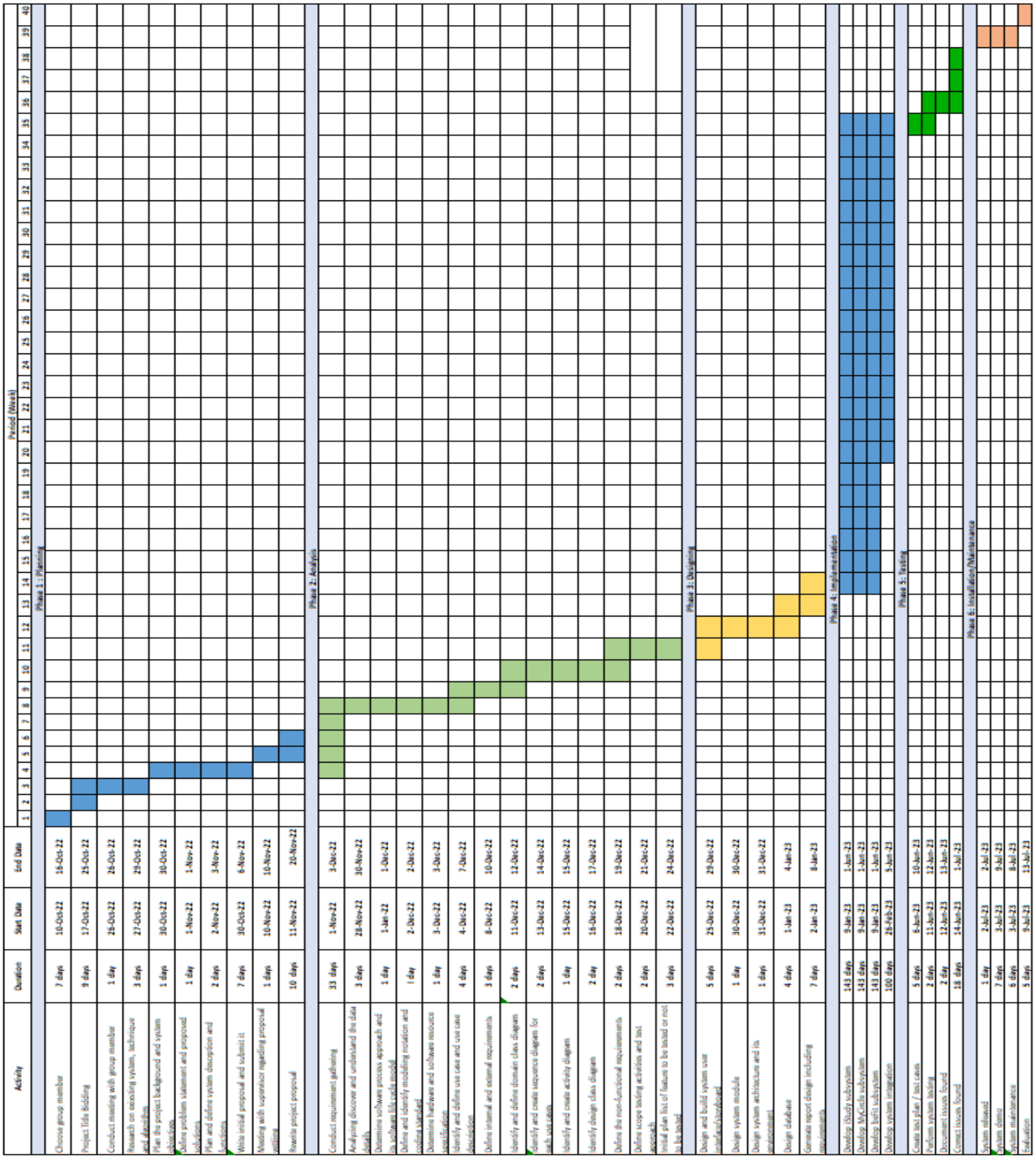


Figure 1.2: *Gantt chart for the overall project schedule*

1.6.1. Hierarchy Chart of Work Division

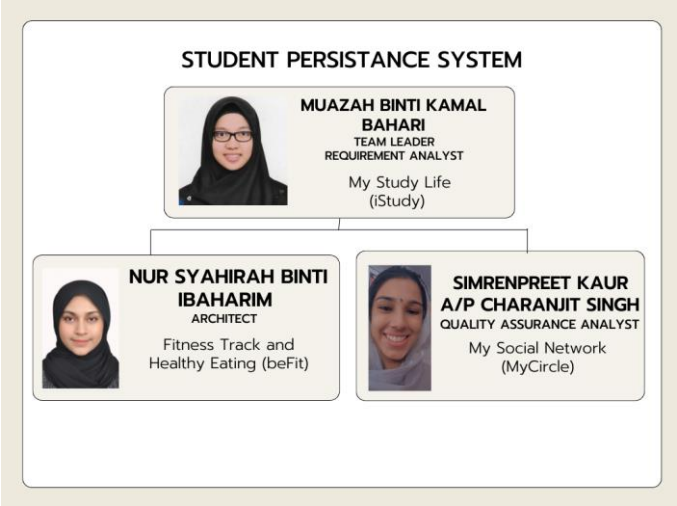


Figure 1.3: Hierarchy Chart

1.6.2. Milestone 1: System Requirement and Design

Activities	Duration (Day)
1. Project management plan	
Conduct a meeting with the group member and supervisor	1 day
Plan project background and determine the organization background	1 day
Define system description and functions	2 days
Determine software process approach and its software life cycle model	1 day
Define and identify modeling notation and coding standard use for this system	1 day
Identify the organizational structure of the system, project team and roles of respective team members as well as development responsibilities	1 day
Determine and describe the computer, hardware, software, and operating system resource specifications	1 day
Create Gantt chart for the development of Student Personal Assistance and hierarchy chart of work division	1 day
Create Work Breakdown Structure (WBS) for each milestone	1 day
Total Days Required	10 days
2. Discover and understand the detail of all aspects of the problem	

Research and analyze the existing systems, and their algorithm	2 days
Analyze the strengths and weaknesses of existing systems	1 day
Define a proposed solution based on the analysis result of existing systems	1 day
Determine and analyze a requirement gathering technique	1 day
Define and identify use cases for this system	2 days
Identify and describe domain model class diagram	2 days
Determine external interfaces requirements	2 days
Determine internal interfaces requirements for iStudy subsystem, MyCircle subsystem and beFit subsystem including use case description, and sequence diagram	7 days
Identify and define the non-functional requirements for this system	2 days
Total Days Required	20 days
3. Design the components of the solution to the problem	
Define and design the storyboard for this system	5 days
Design the system model	1 day
Design the system architecture and its environment (high-level)	1 day
Design and describe the design class diagram	2 day
Design and describe activity diagram	1 day
Total Days Required	10 days
4. Plan for testing activities	
Define the purpose and scope of testing activities	1 day
Define and list the features of the system to be tested	2 days
Identify the features of the system which will not be tested	1 day
Determine the test approach or strategy to be used during this whole project	1 day
Identify and specifying the criteria that will be used to determine whether each test item is pass or fail.	2 days
Total Days Required	7 days

Table 1.5: WBS for Milestone 1

The following team members are responsible for the following tasks in Milestone 1:

Task ID	Responsibility	Remarks
M1-1	All	Project Leader to coordinate individual tasks
M1-2	All	Requirement Analyst help and coordinate the analyze requirement task
M1-3	All	Architect assist design the component or feature part
M1-4	All	Quality Assurance Analyst coordinate the plan for testing activities

Table 1.6: Milestone 1 Task Assignments

1.6.3. Milestone 2: Progress Review (Prototype development)

Activities	Duration (Day)
1. Defining the Concept	
Decide the goals of the project	2 Days
Total Days Required	2 Days
2. Mobile Market Analysis	
Perform competitive analysis	3 Days
Check out the competition and learn from their mistakes	3 Days
Total Days Required	6 Days
3. Define Requirements	
Identify the Key Functionality Requirements	3 Days
Story-mapping and gathering requirements	4 Days
Total Days Required	7 Days
4. Quick Designing	
Create User personas	5 Days
Create User scenarios	5 Days
Craft the primary screens	5 Days
Start wireframing the mobile application	5 Days
Total Days Required	20 Days
5. Building The Prototype	
Create prototype	30

Total Days Required	
6. Testing and Improving	
Test prototype	15 Days
Validate requirements	2 Days
Refine prototype	15 Days
Total Days Required	32 Days

Table 1.7: WBS for Milestone 2

The following team members are responsible for the following tasks in Milestone 2:

Task ID	Responsibility	Remarks
M2-1	All	Identify the problems that are planning to solve
M2-2	All	Conduct a survey to collect qualitative as well as quantitative data
M2-3	Requirement Analyst	List to put on the functionality in the application
M2-4	Architect	Uncover the audience's needs and expectations from the system
M2-5	All	The end result represents the expected app design and functionality
M2-6	Quality Assurance Analyst	Determine minor changes to be implemented, gather some meaningful insights and update the app accordingly.

Table 1.8: Milestone 2 Task Assignments

1.6.4. Milestone 3: Final Presentation (Final development)

Activities	Duration (Days)
1. Deployment	
Decide the environment suitable for deployment and deploy the system.	1 Day
Sprint Retrospective conducted.	1 Day
Feedback gathering from the users.	2 Day

Total Days Required	4 Days
2. Preparation For Final Report Submission.	
Start preparation on final report documentation.	8 Days
Demonstrate to the supervisor the final report contents for feedback.	1 Day
Finalize the final report.	1 Day
Submit the final report.	1 Day
Total Days Required	11 Days
3. Preparation for System Demo Presentation.	
Start preparation on system demo preparation.	4 Days
Demonstrate to the supervisor the system demo presentation contents for feedback.	1 Day
Perform the system demo presentation to the examiners and supervisor.	1 Day
Total Days Required	6 Days

Table 1.9: WBS for Milestone 3

The following team members are responsible for the following tasks in Milestone 3:

Task ID	Responsibility	Remarks
M3-1	All	Decide the environment suitable for deployment.
M3-2	All	Sprint Retrospective conducted.
M3-3	All	Start preparation on final report documentation.
M3-4	All	Demonstrate to the supervisor the final report contents for feedback.

Table 1.10: Milestone 3 Task Assignments

2. Software Requirements Specifications (SRS)

2.1. Background & Related Work

2.1.1. Existing Systems & Algorithms/Theories

There are many systems that can help students to keep track of their academic, social network, and health but all of these are in separate systems. Meanwhile, Student Personal Assistance has all of these subsystems combined and become a one-stop center system which helps the student to manage their academic, social, and health all together by using only one system.

There are several similar existent applications for iStudy in the market, for example, Microsoft Office, My Study Life, and Google Calendar. These applications are digital planners which enable the student to keep track of all of their classes, homework, and projects. Some user prefers to use a system that already has on their laptop which is Microsoft Office. They use Microsoft Office especially Excel and Word to create their own timetables for classes, create task lists manually or use a template that was provided. They need to consistently open and check their created planner in Excel or Word in order to keep track of their timetable and tasks since it lacks a reminder feature. The second existing system is My Study Life which is owned by Emma Whittington and Gabriella Crick Lewis [9]. There are three main features of this application which are it supports rotation timetables as well as weekly schedules. My Study Life also enables users to keep track of all their classes, tasks, assignments, and exams on any device with the auto reminder feature. Users will be able to set reminders and they will receive notifications of upcoming classes and tasks.

Another existing system users always use to keep track of their academic life is Google Calendar. Google Calendar is a calendar management platform that helps users easily schedule tasks like assignments, exams, and events as well as able to receive reminders about forthcoming tasks. Because the calendar is designed for teams, it's simple to share the schedule and make several calendars that users may utilize collectively. Users can utilize search technology which is a technique used in google calendar to look for particular tasks within their calendars. Google Calendar returns a search engine results page (SERP) showing students where specific

scheduled tasks fall on each of their calendars [10]. Additionally, it supports short message service (SMS) which is the format mobile phones use to send text messages [10]. Users can allow Google Calendar to send reminders via SMS to their registered phone number.

There are various social platforms similar to MyCircle that are available for students in the market. One of them is Campus Online USM. Campus Online USM is a platform designed for students at USM. Campus Online USM serves as a comprehensive resource for students to effectively manage their academic and personal responsibilities. It enables students to efficiently handle the various aspects of their student lives. In this discussion, I will focus on the social features of Campus Online USM. Campus Online USM includes a feature for tracking a student's involvement in social clubs and organizations. This feature includes a page where all of the student's participation marks for the events they have attended are displayed, as well as a section listing the positions they hold within their social clubs. This feature is useful for students as it allows them to monitor their social performance and keep track of the participation marks they have earned. The potential downside of Campus Online USM is that it does not have a dedicated platform for fostering student social connections and relationships as the existing system includes all aspects of the student lives into one. Additionally, Campus Online USM does not offer a recommendation feature to help students discover social clubs that align with their interests. The current user interface for Campus Online USM is not visually appealing or interactive, which may make it unenjoyable for students to use.

One of the most notable features in MyCircle is the recommendation system and it is extremely crucial to have an intent recommender algorithm. A good intent algorithm would help the recommendation feature to find social clubs that align with the student interest. There are a number of widely used recommender algorithms available, including collaborative filtering algorithm and clustering algorithm.

Collaborative filtering algorithm is the most common technique that is used when it comes to developing a recommender system that gives better recommendations to the users by gaining more information about user behavior. Collaborative filtering is a method of making recommendations to a user by finding other users who have similar preferences and using their past behaviors to predict what the current user might

like[11]. It involves identifying a group of people with similar tastes and then using the items they have liked to create a ranked list of suggestions for the current user. This algorithm is often used to recommend items or content to users based on their previous behaviors or preferences and can be effective at helping users discover new and relevant content [11]. In short, collaborative filtering is a recommendation system that relies on the past behaviors of similar users to make predictions about what a current user might enjoy. It depends on the choice made, they end up with a collaborative filtering approach where it calculates on the basis of the rating whether it is explicitly where the user specifies dislikes or likes by rating or reacting against any specific product on a scale of 1 to 5 stars or implicitly where user likes and dislikes are recorded and noticed by their actions like clicks, searches, activity records, page views and more. The result of the rating will aid in recommending contents to the user [12]. The downside of collaborative filtering is that collaborative filtering relies on a large amount of data to make recommendations. If a new user or item is introduced, there may not be enough data to make accurate recommendations. [13]Besides that, Collaborative filtering algorithms often struggle with sparse data, where there are a large number of users and items, but only a small number of ratings. In such cases, the algorithms may not be able to find enough similar users or items to make reliable recommendations.[13]

Apart from that, the Clustering Algorithm is also implemented in the recommendation system. It is a frequently employed technique in exploratory data analysis [14]. Clustering involves dividing a dataset into smaller groups, or clusters, based on their similarities. It's a way to identify patterns and relationships in the data by grouping together data points that are similar and separating those that are dissimilar [14]. Thus, it is useful in the recommendation algorithm as it allows for grouping users or items into clusters, and recommendations can then be made based on the cluster that a user or item belongs to. The K-means algorithm is a widely used clustering method. In the K-means algorithm, it divides a dataset into a specified number of clusters. K means clustering starts with a first group of randomly selected centroids which are used to mark the beginning points of every cluster and performs iterative calculations to optimize the position of centroids reassigns data points to the closest centroid until the centroids reach a stable position and the data points are assigned to their respective clusters [14]. Therefore, clustering helps in grouping together similar data points,

which can be useful for finding trends or making predictions. It is not necessarily required to have a history of user behavior in order to perform clustering. Clustering algorithms can be applied to any dataset that has a measure of similarity or distance between data points. That being said, clustering can be particularly useful for analyzing behavior over time, as it can help to identify patterns and trends in the data.

There are several applications that have similar features to beFit such as MyFitnessPal and Strava. MyFitnessPal is a mainly focused calorie tracking app that assists users in keeping track of their daily food and beverage intake by calculating all nutrients, calories, and vitamins [15] while Strava is a social-fitness app and an athlete-focused app that analyses users' performance and, more importantly, keeps track of their training [16].

MyFitnessPal can count how many calories are burned while Strava does not have this feature. MyFitnessPal calculates the calories needed to maintain a user's current weight by first calculating the basal metabolic rate (BMR) and then applying an activity factor to determine total daily energy expenditure (calories). The Harris-Benedict equation can calculate the ideal number of calories (or maintenance calories) to consume if the user wishes to maintain their current body weight. If they want to gain or lose weight, they can use this number as a starting point to eat more or less.

Strava, as a social fitness app, includes an activity feed of all users' friends who follow them. Strava employs a chronological feed ordering option with a random nature of the algorithm, which results in workouts and races being displayed in an unpredictable order and preventing many from being seen. Both MyFitnessPal nor Strava does not have a feature to track steps taken but Strava has a feature to track distance. Strava measures and displays distance when a GPS file is uploaded; it parses the distance data recorded in the file into a data stream to calculate total distance, average speed, and maximum speed. Strava employs a GPS-based, post-upload approach, which means that after GPS data is recorded and uploaded to Strava, it is parsed into data streams and analyzed [17]. At this point, the distance can be calculated using GPS coordinates. Strava uses this method to calculate the distance for any uploaded file that does not include a distance stream. On Strava, distance contributes to the overall distance totals. Furthermore, distance readings contribute to average speed because the average speed is calculated by dividing distance by total moving time.

2.1.2. Strengths and Weaknesses of the Existing Systems

Existing System	Strength	Weakness
Subsystem 1: My Study Life (iStudy)		
Microsoft Office (Excel and Word)	• Ease of use - Easy to use as users are familiar with Excel and Word. • Free template - There are many free templates available on this software.	• Reminder - Lack of a reminder feature that can make users miss the deadline of the tasks. • Time-consuming - Users need to manually organize their own planner which can take so much time. • Respond time slow - If you save a lot of data in this software then using the software becomes slow.
My Study Life	• Reminder - Allow users to create a timetable for classes and receive the reminder for upcoming classes - Allow users to create a task list, set the reminder as well as receive the reminder about the deadline • Ease of use - Easy to navigate and see what is due and when it is due in the dashboard. - Users are able to view a calendar that consists of tasks deadlines, and classes either weekly or monthly.	• Limited feature - Only limited to creating a schedule and task, there is no monitoring academic performance. • Reminder - Unable to set daily reminders every day before an assignment is due without having to manually set separate reminders.
Google Calendar	• Reminder - Allow users to set and receive reminders about the deadline tasks. • Shareable - Can share the schedule with others by the user can either	• Need to pay if want to use more features. - The free version only allows users to share the calendar in read-only form while the paid version allows adding, deleting, or changing events by as many users as possible.

	making their calendar public or sending someone a shareable link to their calendar or adding the individual's email address. • Ease of use - User-friendly and can sync the google calender with other devices by only using the same google account and turning on the auto sync feature	• Internet Connection - While users are offline, changes do not sync with their computer or others' calendars until they are online again.
Subsystem 2: My Social Network (MyCircle)		
Campus Online USM	• Convenient. - Makes students' lives easier by providing everything they need in a single, all-inclusive system. • Comprehensive. - Keeps a detailed log of all the social activities a student is engaged in.	• Interface lacks aesthetic appeal. - The interface has poor color choices, unappealing design elements, and a cluttered layout that makes an interface difficult or unenjoyable to use • Relatively static. - Primarily focused on presenting information, rather than allowing users to actively engage with it.
Instagram	• Visually appealing. - Instagram is known for its focus on visual content that includes photos and videos. Thus, making it an appealing platform for users who want to share and discover visually appealing content • Ease Of Use - Instagram has a simple and intuitive user interface that makes it easy for users to navigate and engage with the platform.	• Limited content posting options. - Instagram only allows users to post photos and videos and does not have other social engagement features with the users. • Causes social isolation. - Instagram is used commonly as a replacement for in-person social interactions.

Subsystem 3: Fitness Track And Healthy Eating (befit)		
MyFitnessPal	• Help users become more mindful of what they are eating - Users can analyse patterns to determine what their diet is lacking or where they should cut back a little. They can also see how much they ate during the day and how many calories are in a particular food or drink. • Convenience - The app allows users to log food from any location and has a food database with over 3,282,000 different types of foods to choose from (customized recipes can be added to the database).	• May stop relying on intuitive eating - Some users may rely solely on the app and the numbers (calories) rather than listening to their bodies' hunger signals. • Can encourage obsession - Users may become overly focused on the numbers (calories) and become neurotic about the diet and numbers.
Strava	• Versatile - The app is versatile in that it supports Bluetooth, ANT devices, and an accurate GPS. It is compatible with both Android and iOS. The premium version includes coaching, live feedback, and a goal monitor, which allows users to enter their desired fitness goals and the app will keep track of their progress. • A friendly community - Users can follow others and encourage them with kudos and comments, making their training more social. They can also join clubs and become involved in the cycling community.	• Best for competition use only - Given that it is primarily a social and athlete-focused app, it may be best to use another app if users are not interested in competing with their workouts. This app is not recommended for personal workouts because of its social component, which users enjoy. • The premium version upgrade is expensive. - It requires users to upgrade to enjoy the fan-favourite features that are expensive.

Table 2.1: Strengths and Weaknesses of the Existing Systems

2.1.3. Problem Summary

Despite the fact that the currently existing systems have favourable features, they still have some limitations and inability to satisfy our target user which is students. Developing Student Personal Assistance with incorporated features that are helpful and practical to students serves as a great solution as it can help the students be more organized, socialized, and healthy. There are many existing systems for students to keep track of their classes and task in the market however it solely focuses on organizing all the tasks that students have to complete during the week, semester, and academic year. It does not offer any other functionalities such as monitoring academic performance or viewing recommended courses. Hence, iStudy offers an academic platform to help students to keep track of their classes, and tasks by notifying or reminding them of the due date. The student also will be able to view academic, social, and health performance whether in the form of a table or chart. The student allows to view an academic performance chart based on grade coursework entered for each course the most important feature is the system can recommend courses for the student.

The existing system that is Campus Online USM does not have a personalized system for students specific for their social lives. It comprises all aspects of student life in it that is extremely static which primarily focuses on presenting information, rather than allowing users to actively engage with it. Thus, it may not be as engaging or valuable as a more dynamic platform. Therefore, MyCircle is suggested that behaves as a platform that allows the student to actively interact with their peers using the available features in the system as well as having the standout feature that is the recommendation system that recommends social clubs to students that align with their social interest. The clustering algorithm is an integral part of the recommendation system in this subsystem, helping to identify and group together similar data points. The attributes used in the clustering algorithm is the student social interest and the existing social clubs there are in USM. Hence, when performing a clustering analysis, the algorithm will use these attributes of the data points to determine how similar or dissimilar they are to each other and will help to group together students with similar social interests towards their interested social clubs. Furthermore, by joining social clubs, students can also participate in social activities that are organized by those

social clubs. Consequently, students can earn participation marks that would be beneficial for the student in being able to build connections with their peers that can be useful to them personally and professionally as well as a platform to release their stress whereby engaging in social activities can provide a break from the demands of academic life and can help students reduce stress and relax. In addition to that, there is also a chat feature that enables students to effortlessly communicate with their peers, fostering a more interactive and engaging environment.

Many companies have created fitness tracking and healthy eating apps. However, most apps only focus on one of the functions, such as fitness tracking or healthy eating, with the main focus varying from one to the other. As a result, users who want to adopt a healthy lifestyle must download two or three apps just to meet their needs and requirements. Users can record their food intake and exercise, but they must do so using two different apps. As a consequence, tracking health in terms of fitness and healthy eating is difficult because each entry is stored in different apps. This is where beFit plays a role as it has both features; fitness tracking and healthy eating to list a few. beFit does not limit itself to one purpose that focuses on only one as it covers both fitness tracking and healthy eating. beFit tracks users' health in fitness and healthy eating by having them input their daily food intake and exercise they have done, similar to a diary, to keep track of their previous exercises and diet. beFit can display a route the users have taken that also tracks how many steps users take throughout the day, allowing them to see how active they have been. The steps taken are then used to calculate the calories burned and the distance travelled accordingly. beFit has this feature that enables the student to plan to exercise together with friends which can help them be physically fit and well-socialized. It also is saved into the diary after it is done to easily keep track of the activities done. beFit also has a feature which rewards the students when they complete goals with the intention to stimulate the students’ motivation to keep striving to adopt a healthy lifestyle.

2.2. Requirements Gathering Techniques

The requirement gathering technique used is a survey technique for requirement gathering purposes and ascertaining the knowledge and requirements of the stakeholders. Using a survey or questionnaire to gather information for our project's requirements is cost-effective and eliminates constraints such as a geographically

dispersed stakeholder base or time constraints, which is especially important given our target audience is students who want to get things done fast. We also get a guarantee of accurate recording of responses because students use phones all the time so they must have preferred mobile apps to ease their life. Given that it is straightforward, the survey or questionnaire leaves no room for interpretation of tone or points that may be lost in translation [18]. We use a survey to collect additional information we needed for the requirements of Student Personal Assistance. The questionnaire has four sections; section 1 is about the respondent's demographic, section 2 for the iStudy subsystem, section 3 for the MyCircle subsystem, and section 4 for the beFit subsystem. This survey was distributed via social media such as *WhatsApp,* and *Telegram.* To elicit requirements from the students, we used open and closed questions (e.g., multiple-choice questions). We use an online questionnaire as a viable option due to the large number of students that can be surveyed. As long as the answers are predetermined, even students who have no prior knowledge or experience with similar apps can explicitly express their knowledge and receive assessments [19].

From the survey, we notice that most of the students found that a planner is useful and important because it can enhance time management skills, improves students' organization skills, and enhances student responsibility and productivity. Microsoft Office such as Excel and Word is the most common use among students to keep track of their classes and tasks but it lacks reminder functionality. Some students also like to use their mobile calendar and google calendar to organize their classes and tasks. The main reason why they use the mobile calendar and google calendar is because it easier to manage and keep track of the task as well as to set reminders for upcoming tasks. Based on the survey students stated that they want an effective reminder feature, a free system, and a user-friendly system.

According to the results of the survey presented in the third section that is for MyCircle, all of the respondents indicated that they use social media. The main two reasons why the respondents like using social media is because of its ease of use along with it helps them to stay updated with the current trends. The majority of respondents cited collecting merit points as their primary reason for joining social clubs at their university, followed by discovering new interests and making new friends. A very small percentage of respondents indicated that their interest in the social club itself

was their primary motivation for joining them. Moving on, the majority of the respondents agreed that it was difficult to find a social club that you have interest in. The reasons were mainly because it has too much workload and difficulty in their time management as well as there are too many social clubs in university that they don't know which one suits them the best. A significant number of survey participants noted that it can be difficult to find friends that share the same interests. Majority of the respondents indicated that outdoor exploration sports would be an activity they would like to do with their friends followed by badminton and hiking.

Based on the survey's findings in the fourth section, many students adopt a healthy lifestyle with the purpose to improve their health. However, not all students use any fitness tracking and healthy eating apps in their life to meet their needs. Only a few students use fitness tracking and healthy eating app like MyFitnessPal with the highest number of respondents, followed by Fitbit and Health app on their phones. The prominent feature that is used most by the students in the app is to track steps and the app chosen is well-liked due to it being easy to use, and convenient to use anywhere and anytime. Some students also reported that they dislike the apps due to the need to subscribe to the premium version to fully utilize all the features that limit all the features available. It is stated from the survey that the students would like to have an app that has features for food nutrition calculation, no need to subscribe to utilize all features and motivation to keep using the app and to adopt a healthy lifestyle. It should be mentioned that the students would like to persevere the motivation going by having rewards in the form of university points which is the Continous Student Development (MyCSD).

2.3. Top Level Representation

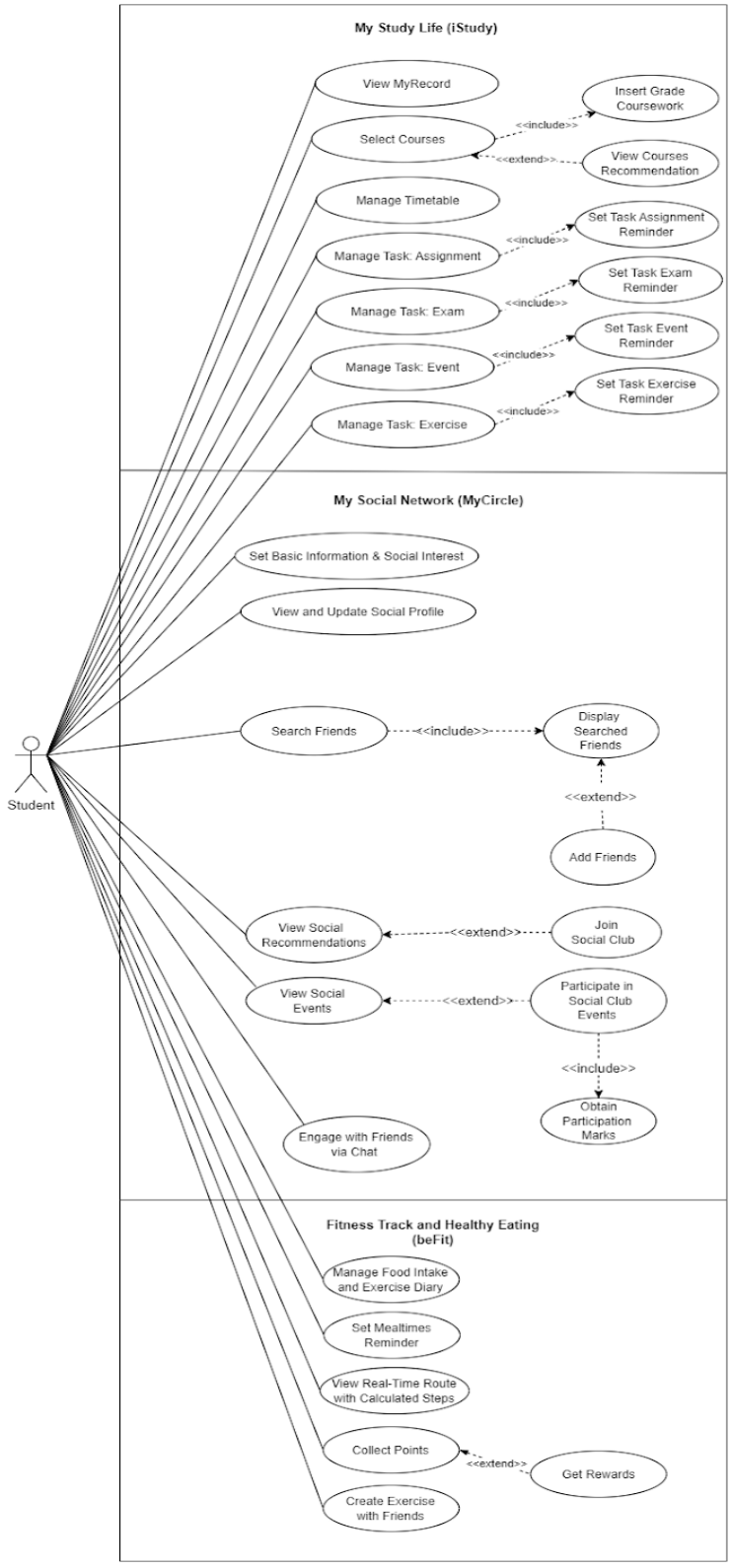


Figure 2.1: Overall Use Case Diagram for Student Personal Assistance

Student Personal Assistance has only 1 actor that interacts with the system that we are modeling which is a student. Subsystem one which is iStudy has seven use cases as shown in Figure 2.1. The checker dashboard module actually represents the entire system which is Student Personal Assistance. This is a landing page for the entire system detailing with navigation, a weekly calendar as well a summary of fitness. The checker dashboard module also includes the first use case which is to View MyRecord aside from viewing the landing page. MyRecord detailing academic, social, and health performance in both table and chart form. Students can view a list of courses taken and their coursework grades for the academic section, for the social section student can view a list of active clubs with their participation marks, and in the health section students are able to view reward points for each goal achieved. The second use case is Select Courses indicating that students can select courses they enroll in throughout the semester and insert the grade coursework so that they can monitor their academic performance chart for each course taken. The list of courses to be selected actually comes from class information in the timetable. When a student creates a new class that consists of a course name, it will appear in the timetable as well as select the course feature. The notable feature of the iStudy subsystem is recommendation courses. The student will be able to view courses recommendation and it is really helpful for those students who want to achieve a good grade but do not know which courses they should enroll in. To be able to use the Select Courses use case, the student should click on Academic Profile in the navigation, hence this makes this use case fall under the academic profiling module. Furthermore use cases such as Manage Timetable, Manage Task: Assignment, Manage Task: Exam, Manage Task: Event, and Manage Task: Exercise fall under the scheduling module. The timetable enables students to manage their classes by allowing the student to create, delete and update their class information such as course name, time, and location in the timetable. Use cases Manage Task: Assignment, Manage Task: Exam, Manage Task: Event, and Manage Task: Exercise work the same. The student can keep track of their task assignments, task exams, task events, and task exercises by allowing them to view, create, update and delete those activities. Being able to manage and keep track of the classes, and tasks are helpful and solve the problem of students missing all of the activities mentioned. iStudy also allows the student to set a reminder and receive a

notification regarding the due date of the tasks. This feature is really helpful for those students who keep forgetting or are not aware of the task's deadline.

The second subsystem is named MyCircle which is built based on 6 use cases. As previously mentioned, this subsystem consists of three main modules, with each use case corresponding to the module it belongs to. The first use case is Set Basic Information and Social Interest belongs to the Student Profiling Module. This use case serves the purpose for the student to set their basic information to be displayed at their social profile and to insert their social interest which will be utilized as an attribute in the clustering algorithm to recommend suitable social clubs for the student to enroll in. Next, the View and Update Social Profile use case also belongs to the Student Profiling Module. This use case serves the purpose to view the student social profile that displays their basic information and their social life where it shows the participation marks collected by the student based on the social events they join in, the number of social clubs they enroll in, and more. Furthermore, this use case allows the student to update their social profile by modifying their basic information and social interests. This ensures that the student's profile accurately reflects their current interests and personal details. The Search Friends use case belongs to the Student Profiling Module. This use case serves the purpose for the student to find friends and add them to their social circle. Moving on, the View Social Recommendation use case is included in the Student Participation Module where in here, the recommendation feature is utilized that it recommends the student social clubs that aligns with their social interest. With this, the student can join social clubs. Additionally, the View Social Events Use Case is included in the Student Participation Module where in here, students are able to view the events held by the social clubs they had joined and participate in these events which consequently enables the student in collecting participation marks. Finally, the Engage With Friends Via Chat Use Case belongs to the Student Socializing Module where students are able to search for friends they want to chat with and chat with them by messaging them via chat. This also includes an Exercise With Friends feature where by clicking it; it will redirect to the Exercise With Friends page. That concludes the Use Cases in MyCircle that will be implemented during the development process.

The third subsystem is beFit which has 5 use cases with each of them having an association link with the actor which is the student. Only 1 base use case has an

extended relationship that adds more functionality to it. The first use case, Manage Food Intake and Exercise Diary is in the Health Tracker Module as it tracks health in terms of fitness tracking and healthy eating. The students can manage the food and exercise– they can add, view, update and delete the data input. The students can input their daily food intake for breakfast, lunch, dinner, and snacks which also include the calories. This is effective to help the students learn about their eating habits, and thus can lead them in making smarter food choices that are rich in nutrition. Students can also input the exercises that they have done for the day to keep track of their activities. The second use case, Set Mealtimes Reminder is used for the students to be reminded of their mealtimes for breakfast, lunch, and dinner so they can eat on time every day. It is also in the Health Module as it helps the students maintain a balanced diet and create a more stable energy source because their metabolism will be working at peak efficiency all day especially when they are busy with many tasks and activities that require energy. The third use case, View Real-Time Route with Calculated Steps is in the beRoute Module as it shows the route taken by the student on a map that also tracks the steps. Thus enables it to calculate the steps taken, distance travelled and calories burned from the walking. The fourth use case, Collect Points has an extended relationship with the Get Rewards use case and it is triggered conditionally. The system has set a list of goals that students can achieve in Food Intake and Exercise Diary, Exercise with Friends and beRoute features so when they complete any goals, they will get points that are collected until they are sufficient to be eligible to get rewards hence triggering the Get Rewards extended use case. These use cases are in the Reward Module as it rewards the students whenever they collect sufficient points to get the rewards. The fifth use case, Create Exercise with Friends is related to the Health Tracker Module as it tracks health in terms of fitness but students will be doing it with friends. The students are required to add or invite their friends to their plan to do exercise together. This will make it more fun, increase their chances of sticking to the exercise plan and be a great way to keep them motivated. The exercise will then be saved into Task: Exercise that has a reminder feature to remind the participants and then after the exercise has been done, it will be saved to the exercise diary that saves all exercises done by the student. View Real-Time Route with Calculated Steps, Create Exercise with Friends and Manage Food Intake and Exercise Diary are in the

Reward Module as well as the students can collect points by completing goals in any of these to get rewards.

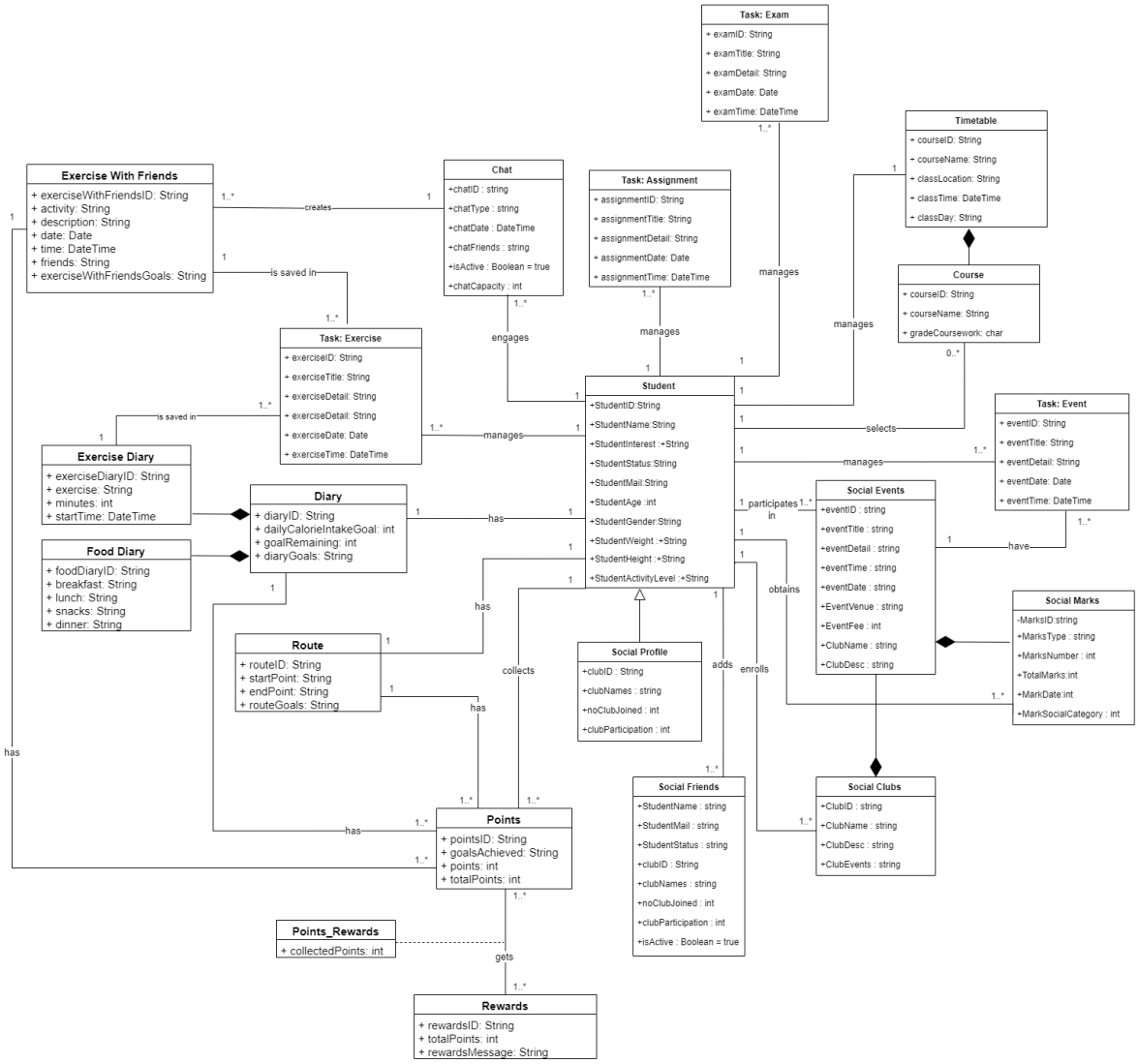


Figure 2.2: Overall Domain Model Class Diagram for Student Personal Assistance

The structures and relationships between the components in an application are graphically represented using domain model class diagrams [20]. For iStudy, a student has a relationship with the Task Assignment table, Task Exam table, Task Event table, Task Exercise table, Timetable table, and Course table. Each student is able to manage many task assignments, task exams, task events, and task exercises that were created by themselves. The reason why these tasks can be managed by only one student is because it is their own task, it does not involve any other student. In each task, there are several values such as title, detail, date, and time. The student also can

view, create, update and delete only their own timetable. The relationship between the student table and the Course table is that student can select many courses they enroll in as well as insert coursework grades they got. The relationship between the Timetable and Course is composition. This is because when the student enters the course name into the timetable, only then the course name will display on the courses to be selected. It means the data such as the course name from the Timetable table will send to the Course table. If the Timetable table is deleted or does not have any class information, then the student would not be able to select any courses which will affect the student cannot insert coursework grades as well as view recommendations based on the courses selected.

For MyCircle, the important table in this subsystem is the Student Class where it has a relationship with every other table. The Student table has a relationship with the Social Profile class where every student that uses this application must have a social profile. The Social Profile table is inherited from the Student Class. The class diagram includes two tables, Student and SocialProfile, which are related to one another through the attributes StudentID and ClubID. The Student table has the attribute StudentInterest and the SocialProfile table has methods such as getStudentInformation and getSocialInformation, which allow for the retrieval of information about the student and their social profile. Next, the class diagram includes a relationship between the Student and SocialClub tables, which represents the enrollment of students in various social clubs at the university. The SocialClub table stores attributes such as ClubName and ClubID, as well as methods such as getClubInformation and joinClub. These attributes and methods allow for the management and tracking of students who are interested in joining a particular social club. The relationship between the Student and Social Clubs tables depicts the enrollment of students in the clubs that they are interested in. The Student has a relationship with the Social Events to depict the relationship of students participating in the Social Events which then indirectly reflects the relationship between the Student and Social Marks where students gain participation marks by participating in the social club events. Moving on, the Social Events table has a composition relationship with the Social Clubs class where the Social Events class is composed of one or more instances of the Social Clubs table, and the Social Events instance is destroyed when the Social Clubs instance is destroyed. Other than that, the Social Marks table also has a composition

relationship with the Social Events table where the Social Marks table is composed of one or more instances of the Social Events table, and the Social Marks instances are destroyed when the Social Events instance is destroyed. The social event joined will then be saved into the Task: Event table. The Student table has a relationship with the Social Friend table to depict the relationship of the student adding their social friends into their social circle. The Student table has a relationship with the Chat table to depict the engagement within students via Chat.

For beFit, it presents how data from a student can be converted into a complete set of information to use as it communicates what business processes are included in the application. From the diagram, it shows the student has a diary with a composition relationship with the food diary and exercise diary to input their daily food intake and exercise done that also able to be managed by the student. The student can manage their own diary. If the Diary table is deleted or non-functional, both Food Diary and Exercise Diary tables will also be deleted or non-functional. The student can view a route on a map that has a start point and end point as it has an association relationship between the tables Student and Route. The student can create many Exercise with Friends that will be able to add friends to their plan to do exercise together; from the tables Student and Exercise with Friends have an association relationship. The list of friends that can be added is retrieved from the table Chat. The exercises with friends created will then be saved into the Task: Exercise table. The student along with the friends that have been added can view the exercise(s) with friends that is/are saved in the task exercise. The Task: Exercise table gets the data from the unique ID to display all the information in the task exercise as the exercise with friends is saved in it. All Diary, Route and Exercise with Friends tables have the system set the goals for the student to complete. This is crucial for the Points table as it gets the points that are collected by the student after they complete the goals as they have an association connection. Tables Points and Rewards have an association connection between them as the Reward table gets the points collected to reckon whether the student gets the reward or not.

2.4. External Interfaces Requirements

2.4.1. Interface [Student/ Student Personal Assistance]

Subsystem 1: My Study Life (iStudy)		
Identifier	**Description**	**Association**
REQ-001	This actor is a student that is able to view their academic, social, and health performance in both table and chart form.	- View MyRecord
REQ-002	This actor is a student who use this system to view, and select courses, insert grade coursework as well as view courses recommendation.	- Select courses - Insert grade coursework - View courses recommendation
REQ-003	This actor is a student who uses this system to view, create, update and delete classes into the timetable.	- Manage timetable
REQ-004	This actor is a student that uses this system to keep track and manage their task assignment. Students are also allowed to set and receive reminders.	- Manage task: Assignment - Set task: Assignment reminder
REQ-005	This actor is a student who uses this system to manage and monitor their task exam as well as set and receive a reminder.	- Manage task: Exam - Set task: Exam reminder
REQ-006	This actor is a student that uses this system to keep track and manage their task event. Students are also allowed to set and receive reminders so they would not forget the date.	- Manage task: Event - Set task: Event reminder
REQ-007	This actor is a student who uses this system to manage and monitor the task exercise they plan to do as well as set and receive a reminder.	- Manage task: Exercise - Set task: Exercise reminder
Subsystem 2: My Social Network (MyCircle)		
Identifier	**Description**	**Association**
REQ - 008	The actor is a student that is able to set their basic information and social interest to be displayed at their profile page.	-Set Basic Information and Social Interest
REQ-009	The actor is a student that is able to view their participation marks for club activities and update their basic information and social interest.	-View and Update Social Profile

REQ-010	The actor is a student that is able to view a list of other students using this application, search for friends to add, and option to add them as friends.	- Search Friend - Display Searched Friends - Add Friends
REQ-011	The actor is a student that is able to view social clubs that are recommended to them based on their interest input and optioned to join clubs that they are interested in.	-View Social Recommendations - Join Social Clubs.
REQ-012	The actor is a student that is able to view social events held and participate in events that they are interested in; consecutively obtaining participation marks from it.	- View Social Events. - Participate in Social Events - Obtain Participation Marks
REQ-013	The actor is a student that is able to engage with their friends socially via chat.	- Engage with Friends via Chat
Subsystem 3: Fitness Track and Healthy Eating (beFit)		
Identifier	**Description**	**Association**
REQ-014	The actor is a student that can manage their food intake and exercise diary by inputting their daily food intake and exercises they have done.	- Manage Food Intake and Exercise Diary
REQ-015	The actor is a student that can set reminders for mealtimes for breakfast, lunch and dinner.	- Set Mealtimes Reminder
REQ-016	The actor is a student that can view their route taken with the calculated steps, distance travelled and calories burned	- View Real-Time Route with Calculated Steps
REQ-017	The actor is a student that can collect points after completing goals to get rewards	- Collect Points - Get Rewards
REQ-018	The actor is a student that creates plans to do exercises together with friends.	- Create Exercise with Friends

Table 2.2: List of Requirements

2.5. Internal Interfaces Requirements for My Study Life (iStudy)

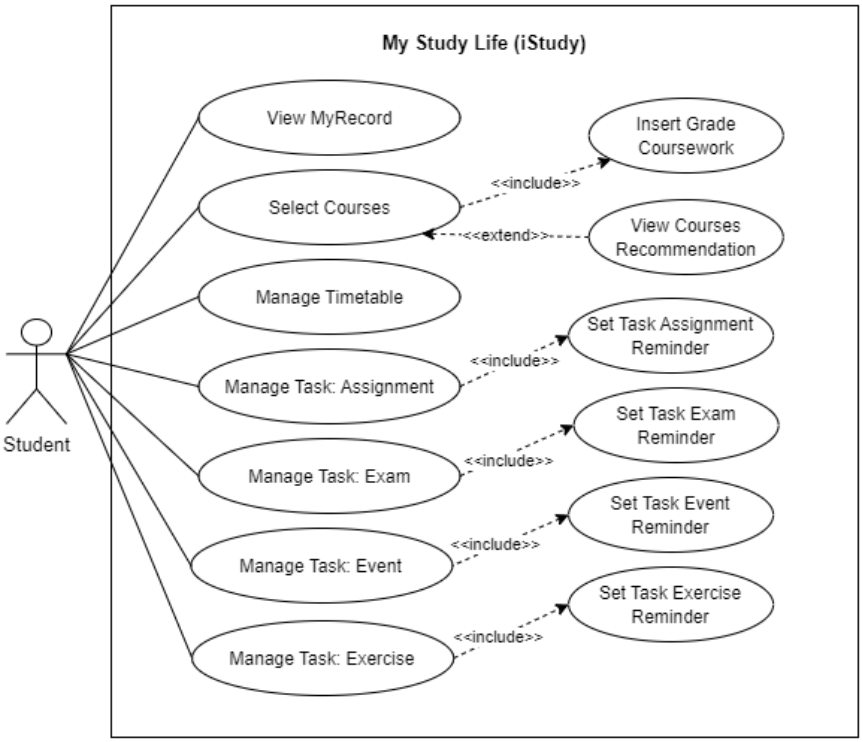


Figure 2.3: Use Case Diagram for iStudy Subsystem

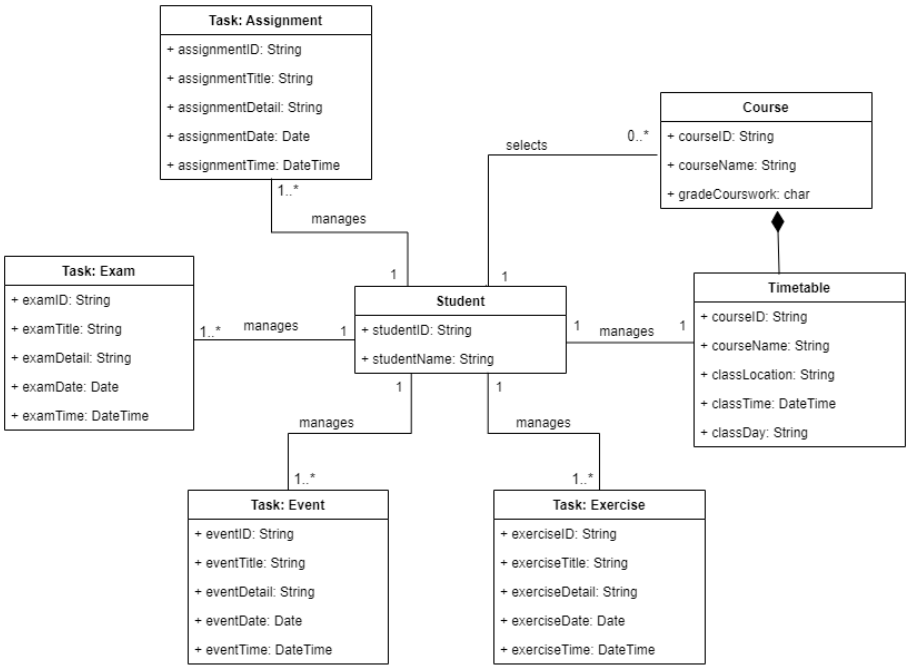


Figure 2.4: Domain Class Diagram for iStudy Subsystem

2.5.1. Use-Case: View MyRecord

2.5.1.1 Identifier: [SRS-001]

2.5.1.2 Use Case Description

Use Case Name:	View MyRecord	
Scenario:	Display academic, social, and health performance.	
Triggering Event:	The student wants to view all of their academic, social, and health performance on the same page	
Brief Description:	The student views their academic, social, and health performance in both table and chart form. For example, a list of courses taken and its coursework grades. The student also wants to view a list of active clubs and their participation marks as well as the health goal list and its rewards points.	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Students need to be logged in into the application. Students need to have courses taken data, active clubs data, and health activities data.	
Postconditions:	Students are able to view their academic, social, and health performance in form of a table and chart.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ001-A1]	1. The student clicks the "MyRecord" page.	1.1 System display "MyRecord" page.
[REQ001-A2]	2. Student views academic, social, and health performance in form of tables and also charts.	2.1 System fetches academic, social, and health information from the database.
Exception Flow: [SRS-001-E2.1]	2.1 If the student does not have data for courses, active clubs, and health activities, then the system would not display it.	

Table 2.3: Use Case Description for View MyRecord

2.5.1.3 System Sequence Diagram

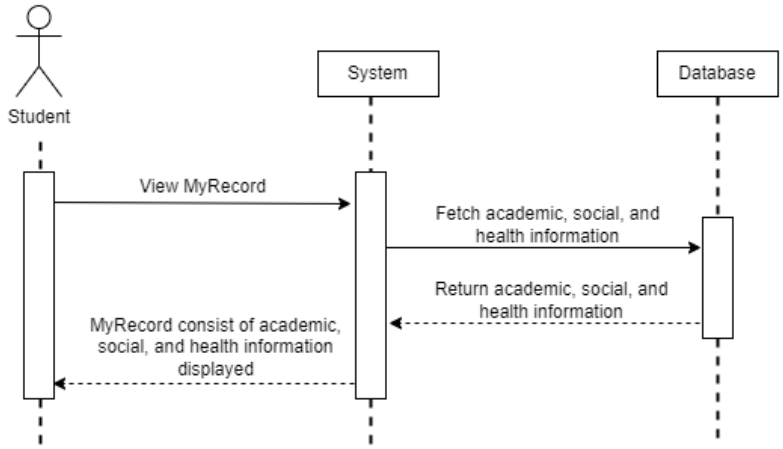


Figure 2.5: Sequence Diagram for View MyRecord

2.5.2. Use-Case: Select Courses

2.5.2.1 Identifier: [SRS-002]

2.5.2.2 Use Case Description

Use Case Name:	Select Courses	
Scenario:	Select courses to enroll.	
Triggering Event:	The student selects the courses to enroll in.	
Brief Description:	Student selects the courses they enroll in throughout the semester and inserts coursework grades they got. They also want to view the list of course recommendations.	
Actors:	Student	
Related Use Cases:	Include: Insert Grade Coursework Extend: View Courses Recommendation	
Preconditions:	The student needs to be logged in to the application. The student needs to have class information in their timetable.	
Postconditions:	Students are able to select the course they enroll in, insert its grade coursework, and view courses recommendation.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ002-A1] [REQ002-A2] [REQ002-A3]	1. Students click “Academic Profile” to be able to select courses. 2. The student selects the course they enroll in and inserts the coursework	1.1 System will display the “Academic Profile” page detailing the list of courses taken with its grade coursework. 2.1 System saves data to the database.

[REQ002-A4]	grades. 3. Student view recommended courses. 4. The student also can update their coursework grade if they want.	2.2 System displays selected course and its coursework grades. 2.3 System displays the academic performance in the form of charts. 3.1 The system fetches and displays the recommended course list from the database. 4.1 Update to the database corresponds to the changes made by the student. 4.2 System displays the updated coursework grade. 4.3 System displays the updated academic performance in the form of charts.
Exception Flow: [SRS-002-E3.1]	3.1 The student would not be able to view recommended courses if there is no selected course.	

Table 2.4: Use Case Description for Select Courses

2.5.2.3 System Sequence Diagram

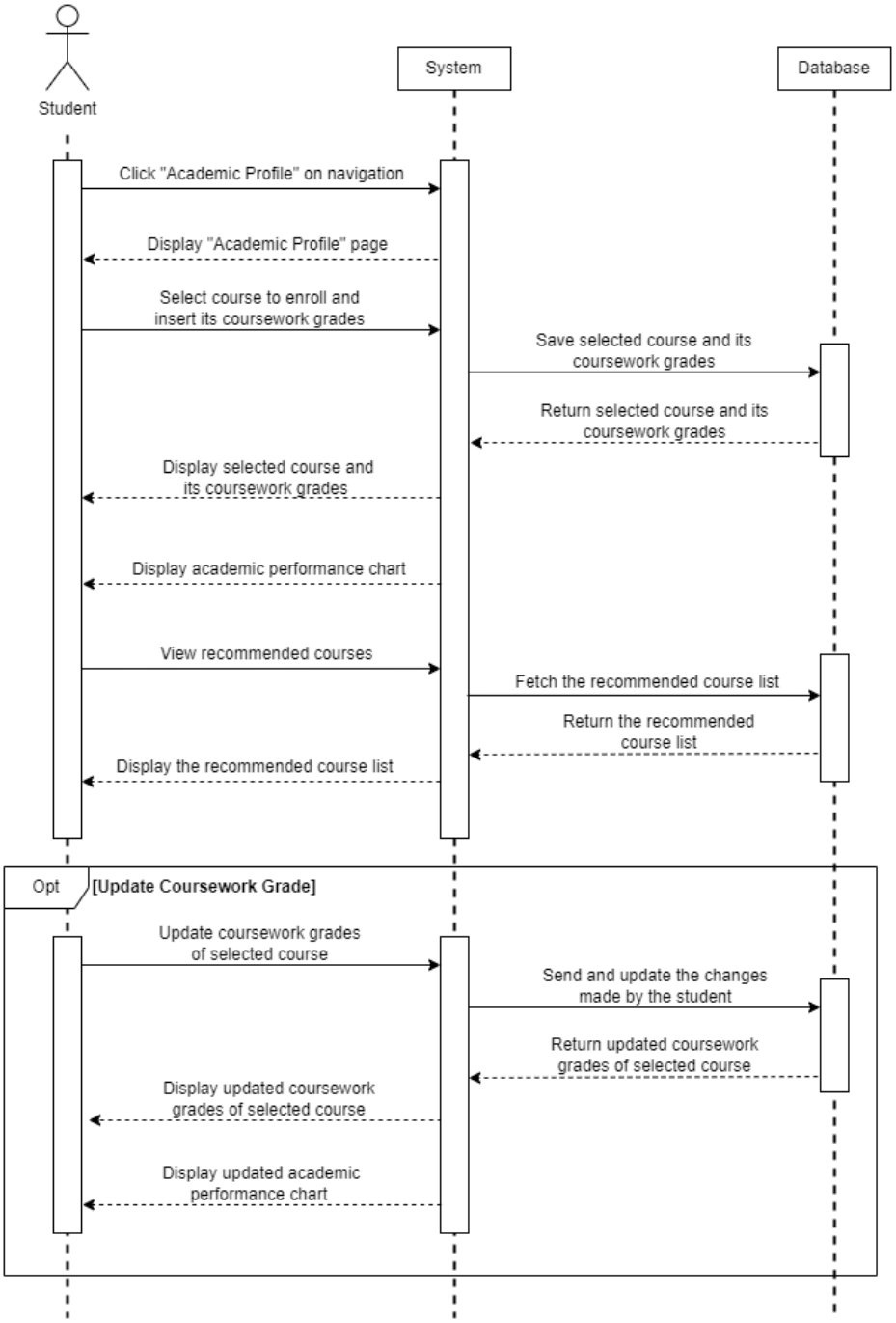


Figure 2.6: Sequence Diagram for Select Courses

2.5.3. Use-Case: Manage Timetable

2.5.3.1 Identifier: [SRS-003]

2.5.3.2 Use Case Description

Use Case Name:	Manage Timetable	
Scenario:	View, create, update and delete the class information in the timetable for a certain day.	
Triggering Event:	The student wants to manage and keep track of their class timetable	
Brief Description:	The student wants to create a new class in the timetable by entering the course name, time, and location into the day chosen. Students also want to be able to update and delete the class information in the timetable.	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Students need to be logged in to the application.	
Postconditions:	Students are able to view class information in the timetable. Students are able to create, update and delete the class in the timetable.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ003-A1]	1. The student clicks the "Timetable" page.	1.1 System display ¨Timetable¨ page
[REQ003-A2]	2. Student views the timetable based on the day the student chooses.	2.1 Fetch class information in the timetable
[REQ003-A3]		2.2 The system displays the list of classes in the timetable based on the day the student chooses.
[REQ003-A4]		
[REQ003-A5]	3. Student creates a new class in the timetable by inputting the course name, time, and location into the chosen day.	3.1 The system will save those information in the database.
	4. Student update the class information in the timetable.	3.2 System displays a new class created in the timetable.
		4.1 System will update the class information in the timetable in the database and save it.
	5. Student deletes the class from the timetable.	4.2 System displays the updated class information in the timetable.

		5.1 The system will delete the chosen class from the timetable from the database.
Exception Flow: [SRS-003-E4.1] [SRS-003-E5.1]	4.1 Student would not be able to update class information if there are no prior classes in the timetable have been created. 5.1 Student would not be able to delete any class if there is no prior classes in the timetable has been created.	

Table 2.5: Use Case Description for Manage Timetable

2.5.3.3 System Sequence Diagram

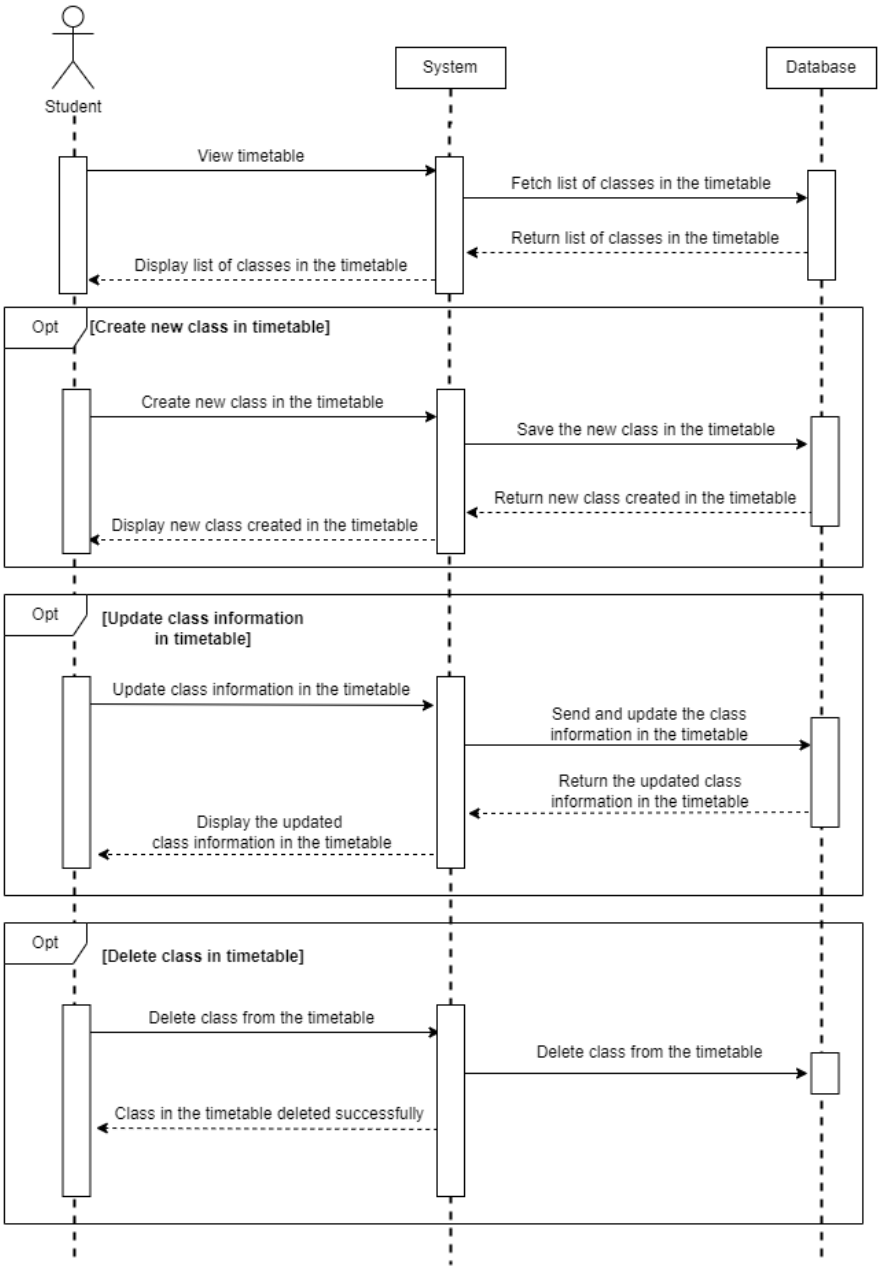


Figure 2.7: Sequence Diagram for Manage Timetable

2.5.4. Use-Case: Manage Task: Assignment

2.5.4.1 Identifier: [SRS-004]

2.5.4.2 Use Case Description

Use Case Name:	Manage Task: Assignment	
Scenario:	Manage task under the category Assignment	
Triggering Event:	Student wants to manage and keep track of their task assignment	
Brief Description:	The student wants to view, update and delete the task assignment they have saved. The student also wants to create a new task assignment. The student also wants to set the task assignment reminder as well as receive the notification of the task assignment deadline.	
Actors:	Student	
Related Use Cases:	Include: Set Task Assignment Reminder	
Preconditions:	Students need to be logged in to the application.	
Postconditions:	The student can manage the task assignment they have created. The student can set reminders and be notified of the deadline. The assignment has been created and saved under the category 'Assignment'.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ004-A1]	1. The student clicks the "Task" page.	1.1 System display ¨Task¨ page
[REQ004-A2]	2. The student chooses a category Assignment and views the task assignment they have created.	2.1 Fetch the list of the task assignment created. 2.2 The system displays the list of the task assignment created.
[REQ004-A3]	3. The student creates a new task assignment by inputting the title, detail, date, and time to set the reminder.	3.1 The system will save those information in the database. 3.2 The system displays a new task assignment. 3.3 The system will notify the student of the deadline for the task assignment.
[REQ004-A5]	4. Student receives a reminder of the task assignment deadline.	
[REQ004-A6]	5. Student update the task assignment information.	5.1 Update to the database corresponds to the changes made by the student and save it. 5.2 The system displays the updated task assignment information
	6. Student deletes any task assignment.	6.1 The system will delete the task assignment from the database.
Exception Flow:	4.1 If the student does not set the reminder then they would not	

[SRS-004-E4.1] [SRS-004-E5.1] [SRS-004-E6.1]	receive the reminder. 5.1 The student would not be able to update task assignment information if there is no prior task assignment has been created. 6.1 Student would not be able to delete task assignment if there is no prior task assignment has been created.

Table 2.6: Use Case Description for Manage Task: Assignment

2.5.4.3 System Sequence Diagram

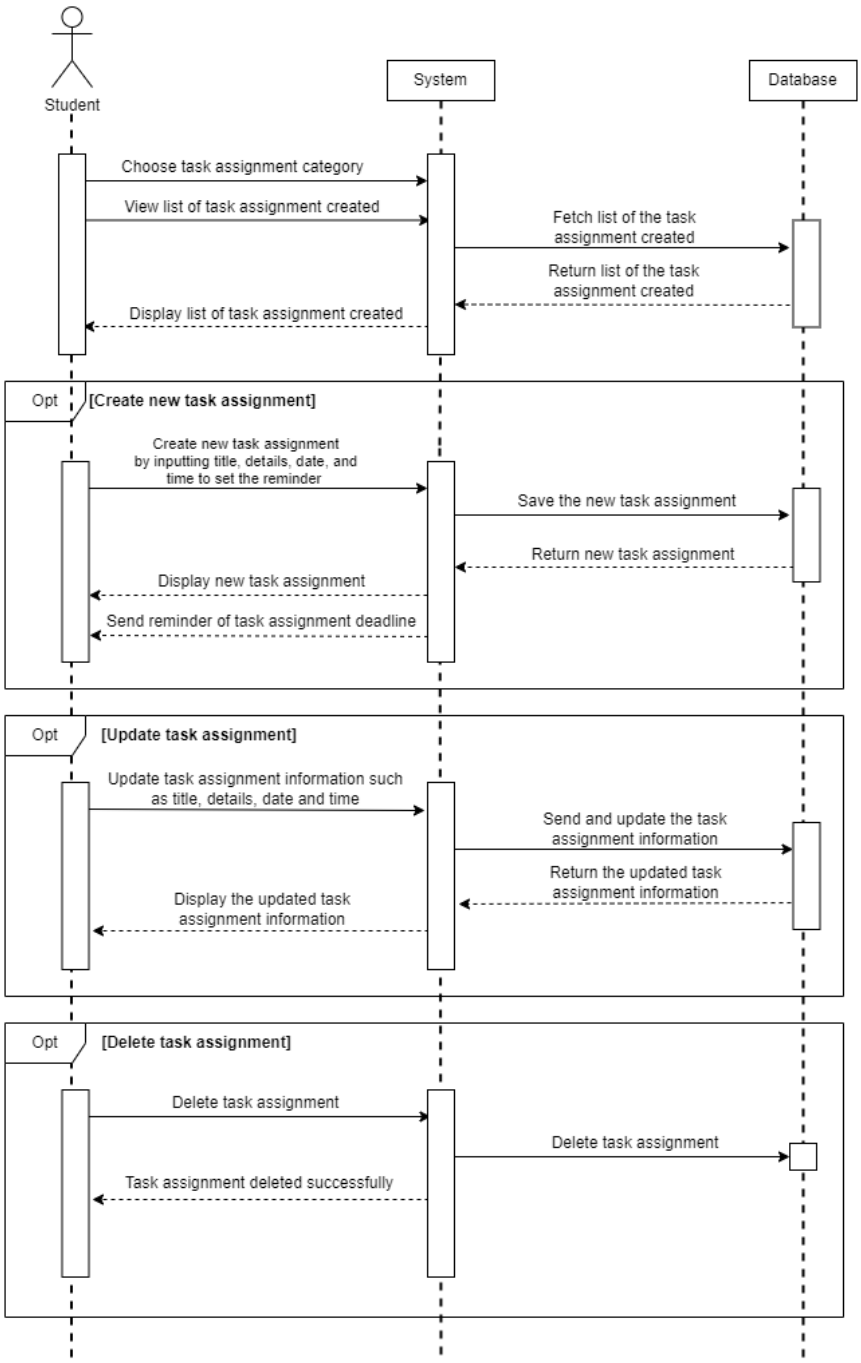


Figure 2.8: Sequence Diagram for Manage Task: Assignment

2.5.5. Use-Case: Manage Task: Exam

2.5.5.1 Identifier: [SRS-005]

2.5.5.2 Use Case Description

Use Case Name:	Manage Task: Exam	
Scenario:	Manage task under category Exam	
Triggering Event:	Student wants to manage and keep track of their task exam	
Brief Description:	The student wants to view, update and delete the task exam they have saved. The student also wants to create a new task exam. The student also wants to set the task exam reminder as well as receive the notification of the task exam date.	
Actors:	Student	
Related Use Cases:	Include: Set Task Exam Reminder	
Preconditions:	The student needs to be logged in into the application.	
Postconditions:	The student can manage the task exam they have created. The student can set a reminder and be notified of the date. The exam has been created and saved under the category Exam..	
Normal/Alternate Flow:	**Actor**	**System**
[REQ005-A1] [REQ005-A2] [REQ005-A3] [REQ005-A5] [REQ005-A6]	1. The student clicks the “Task” page. 2. The student chooses the category Exam and views the task exam they have created. 3. The student creates a new task exam by inputting the title, detail, date, and time to set the reminder. 4. The student receives a reminder of the task exam date. 5. Student update the task exam information. 6. The student deletes any task exam.	1.1 System display ¨Task¨ page 2.1 The system fetches the list of the task exam created 2.2 The system displays the list of the task exam created. 3.1 The system will save those information in the database. 3.2 The system displays a new task exam. 3.3 The system will notify the student of the task exam date. 5.1 Update to the database corresponds to the changes made by the student and save it. 5.2 The system displays the updated task exam information 6.1 The system will remove the task exam from the database

Exception Flow: [SRS-005-E4.1] [SRS-005-E5.1] [SRS-005-E6.1]	4.1 If the student does not set the reminder then they would not receive the reminder. 5.1 The student would not be able to update task exam information if there is no prior task exam has been created. 6.1 The student would not be able to delete a task exam if there is no prior task exam has been created.

Table 2.7: Use Case Description Manage Task: Exam

2.5.5.3 System Sequence Diagram

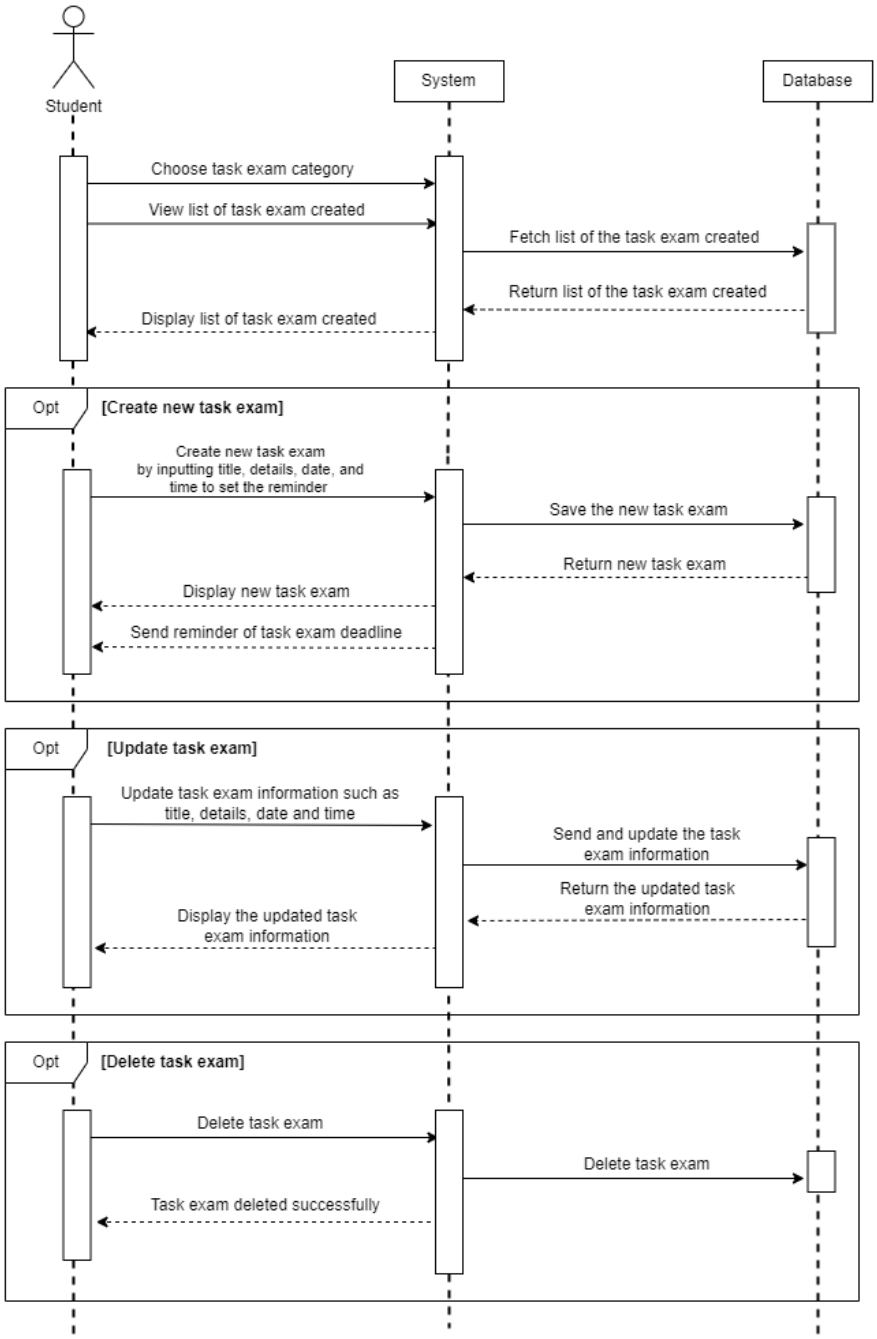


Figure 2.9: Sequence Diagram for Manage Task: Exam

2.5.6. Use-Case: Manage Task: Event

2.5.6.1 Identifier: [SRS-006]

2.5.6.2 Use Case Description

Use Case Name:	Manage Task: Event	
Scenario:	Manage task under category Event	
Triggering Event:	Student wants to manage and keep track of their task event	
Brief Description:	The student wants to view, update and delete the task event they have saved. The student also wants to create a new task event. The student also wants to set the task event reminder as well as receive the notification of the task exam date.	
Actors:	Student	
Related Use Cases:	Include: Set Task Event Reminder	
Preconditions:	The student needs to be logged in into the application.	
Postconditions:	The student can manage the task event they have created. The student can set a reminder and be notified of the date. The event has been created and saved under the category Event.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ006-A1] [REQ006-A2] [REQ006-A3] [REQ006-A5] [REQ006-A6]	1. The student clicks the “Task” page. 2. The student chooses the category Event and views the task event they have created. 3. The student creates a new task event by inputting the title, detail, date, and time to set the reminder. 4. The student receives a reminder of the task event date. 5. Student update the task event information. 6. The student deletes any task event.	1.1 System display ¨Task¨ page 2.1 The system fetches the list of the task event created 2.2 The system displays the list of the task event created. 3.1 The system will save those information in the database. 3.2 The system displays a new task event. 3.3 The system will notify the student of the date for the task event. 5.1 Update to the database corresponds to the changes made by the student and save it. 5.2 The system displays the updated task event information. 6.1 The system will remove the task event from the database
Exception Flow: [SRS-006-E4.1] [SRS-006-E5.1] [SRS-006-E6.1]	4.1 If the student does not set the reminder then they would not receive the reminder. 5.1 The student would not be able to update task event information if there is no prior task event has been created.	

	6.1 The student would not be able to delete any task event if there is no prior task event has been created.

Table 2.8: Use Case Description for Manage Task: Event

2.5.6.3 System Sequence Diagram

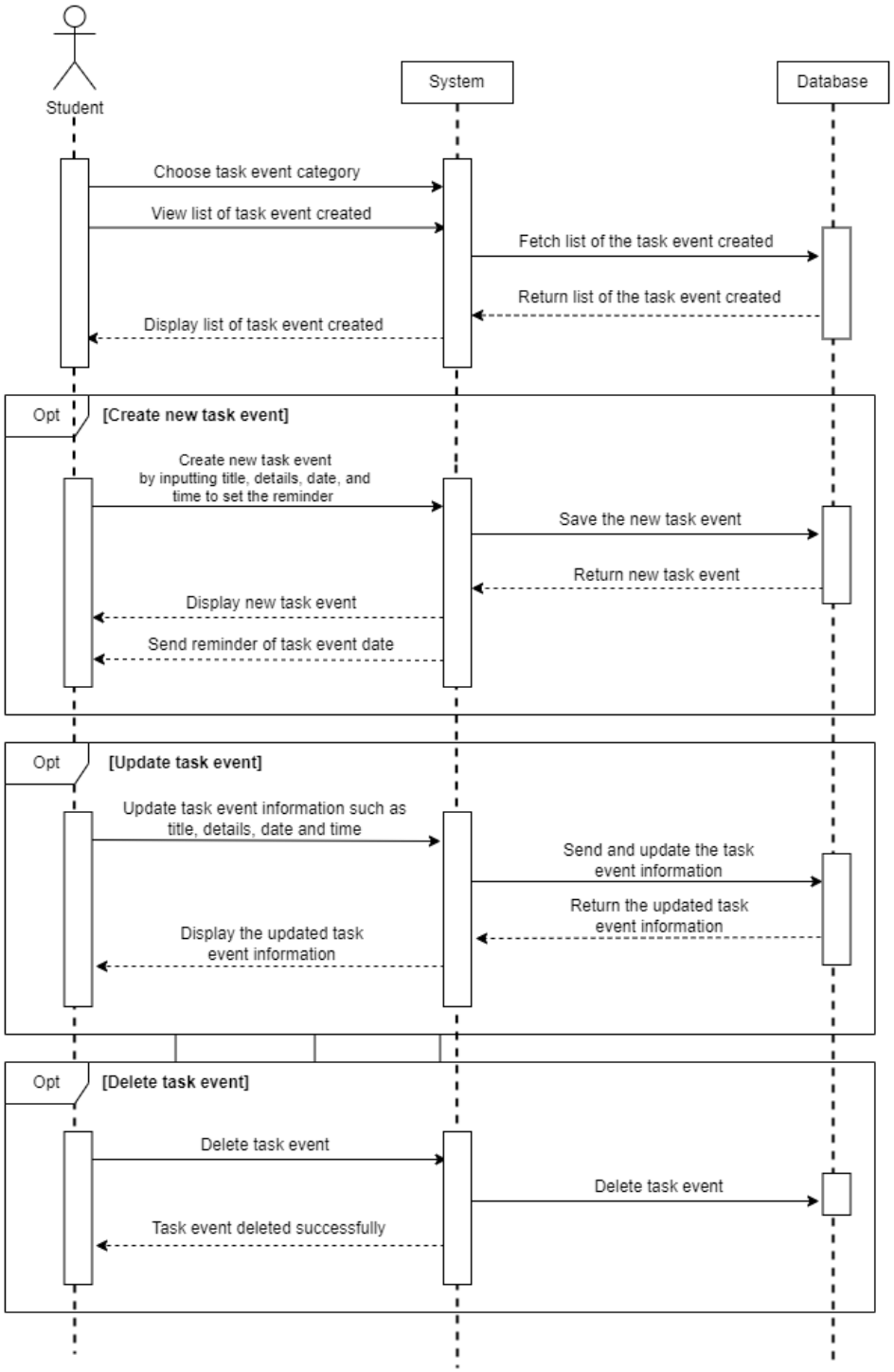


Figure 2.10: Sequence Diagram for Manage Task: Event

2.5.7. Use-Case: Manage Task: Exercise

2.5.7.1 Identifier: [SRS-007]

2.5.7.2 Use Case Description

Use Case Name:	Manage Task: Exercise	
Scenario:	Manage task under category Exercise	
Triggering Event:	Student wants to manage and keep track of their task exercise	
Brief Description:	The student wants to view, update and delete the task exercise they have saved. The student also wants to add a new task exercise. The student also wants to set the task exercise reminder as well as receive the notification of the task exercise date.	
Actors:	Student	
Related Use Cases:	Include: Set Task Exercise Reminder	
Preconditions:	The student needs to be logged in to the application.	
Postconditions:	The student can manage the task exercise they have created. The student can set a reminder and be notified of the date. The exercise has been created and saved under the category Exercise.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ007-A1] [REQ007-A2] [REQ007-A3] [REQ007-A5] [REQ007-A6]	1. The student clicks the “Task” page. 2. The student chooses the category Exercise and views the task exercise they have created. 3. The student creates a new task exercise by inputting the title, detail, date, and time to set the reminder. 4. The student receives a reminder of the task exercise date. 5. Student update the task exercise information. 6. Student deletes any task exercise.	1.1 System display ¨Task¨ page 2.1 The system fetches the list of the task exercise created 2.2 The system displays the list of the task exercise created. 3.1 The system will save those information in the database. 3.2 The system displays a new task exercise. 3.3 The system will notify the student of the date for the task exercise. 5.1 Update to the database corresponds to the changes made by the student and save it. 5.2 The system displays the updated task exercise information 6.1 The system will remove the task exercise from the database
Exception Flow: [SRS-007-E4.1] [SRS-007-E5.1]	4.1 If the student does not set the reminder then they would not receive the reminder. 5.1 The student would not be able to update task exercise	

[SRS-007-E6.1]	information if there is no prior task exercise has been created. 6.1 The student would not be able to delete any task exercise if there is no prior task exercise has been created.

Table 2.9: Use Case Description for Manage Task: Exercise

2.5.7.3 System Sequence Diagram

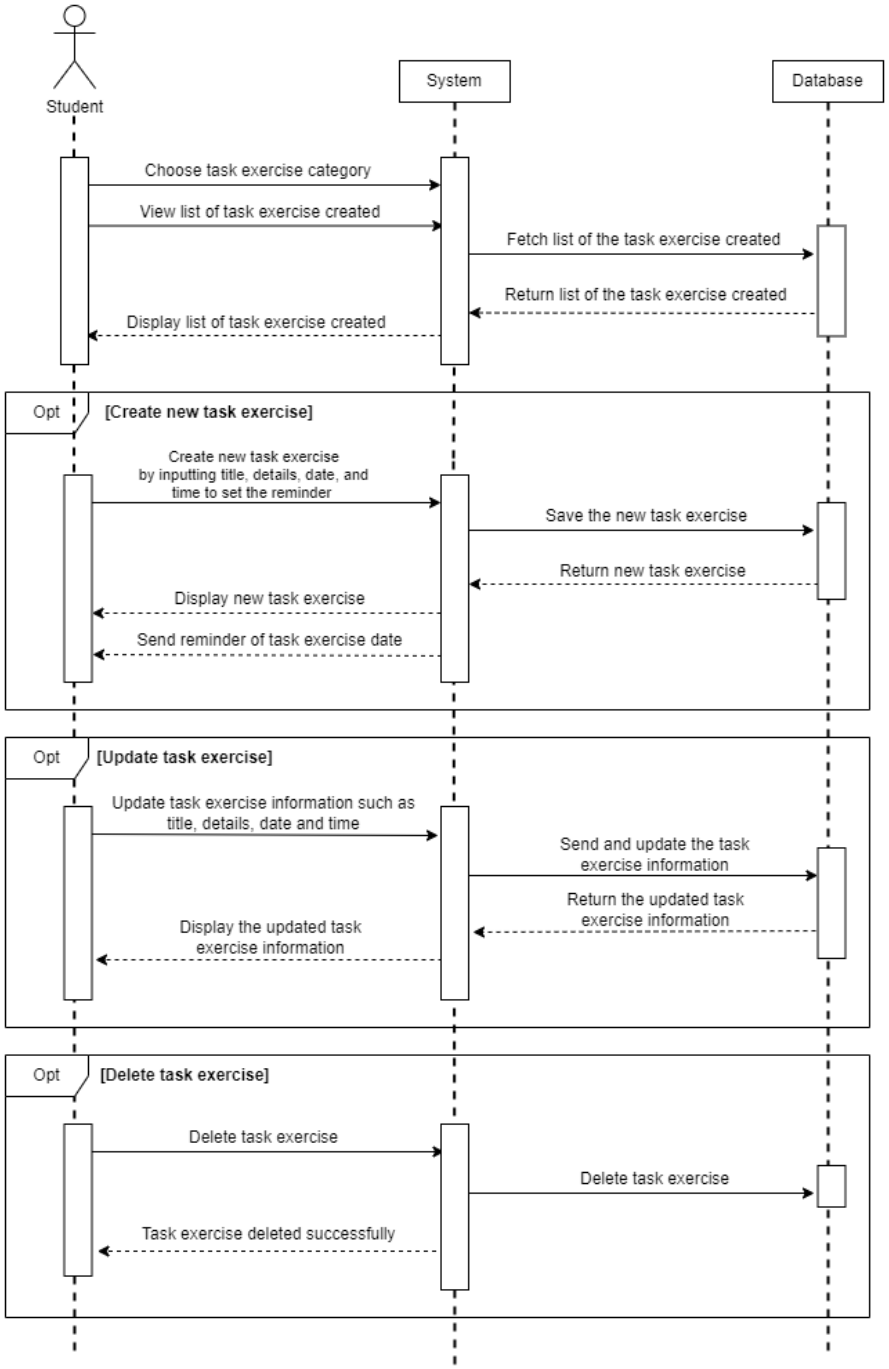


Figure 2.11: Sequence Diagram for Manage Task: Exercise

2.6. Internal Interfaces Requirements for My Social Network (MyCircle)

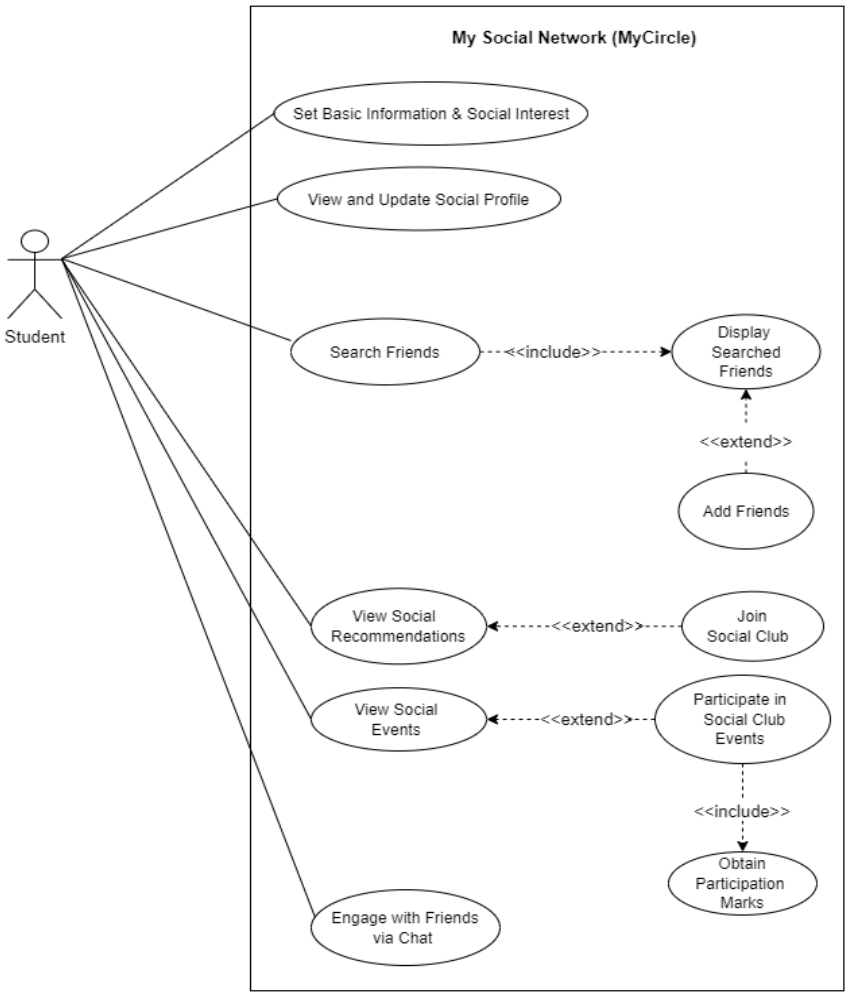


Figure 2.12: Use Case Diagram for MyCircle Subsystem

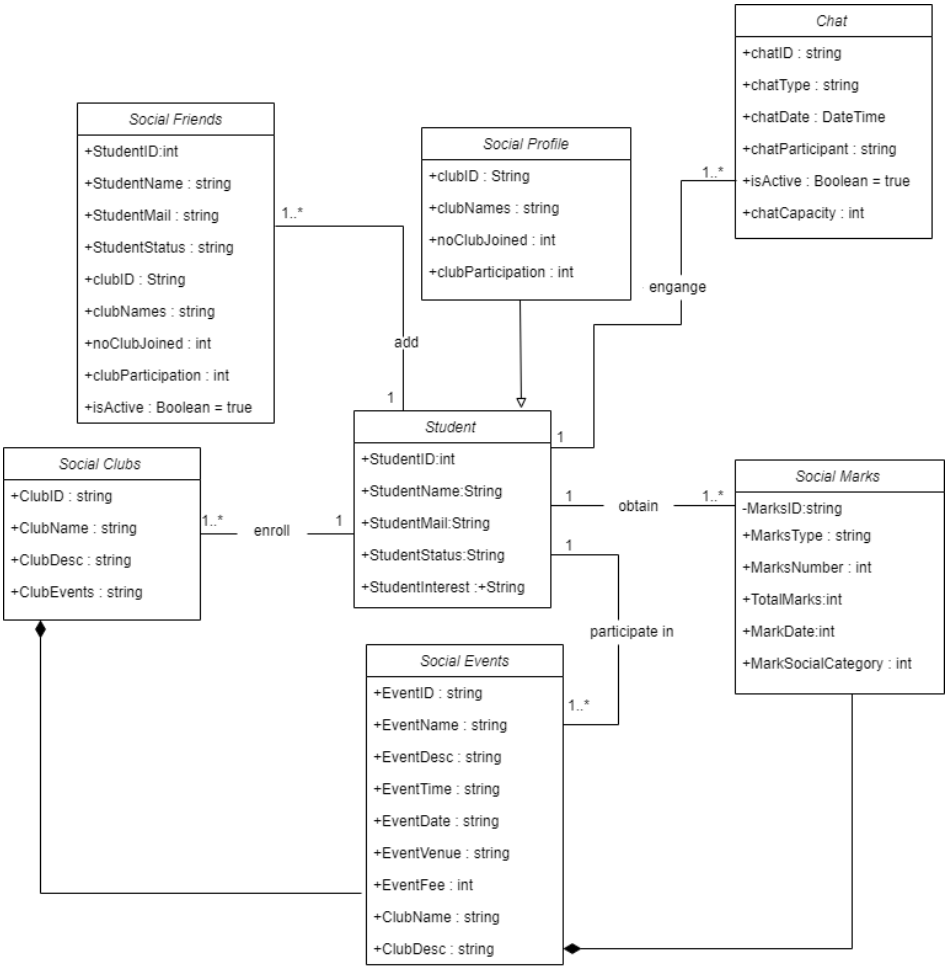


Figure 2.13: Domain Class Diagram for MyCircle Subsystem

2.6.1. Use-Case: Set Basic Information and Social Interest.

2.6.1.1 Identifier: [SRS-008]

2.6.1.2 Use Case Description

Use Case Name:	Set Basic Information and Social Interest.	
Scenario:	Set student basic information and social interest to be displayed on their social profile.	
Triggering Event:	The student wants to set their basic information and social interests.	
Brief Description:	As a student, be able to set their basic information and social interest.	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Students have logged into the system.	
Post conditions:	Student can insert their basic information and social interest and it is being saved and displayed at their social profile page.	
Normal/Alternate Flow:	**Actor**	**System**

[REQ008-A1] [REQ008-A2] [REQ008-A3]	1. Social Information Form is displayed. 2. The student fills in the social information form. 3. Student basic information is displayed at their social profile page.	1.1 System displays the social information form. 2.1 System saves the inputs from student into the database. 3.1 System displays the basic information on their social profile page.
Exception Flow:	The system is offline or inaccessible. The database is offline or inaccessible.	

Table 2.10: Use Case Description for Set Basic Information and Social Interest

2.6.1.3 System Sequence Diagram

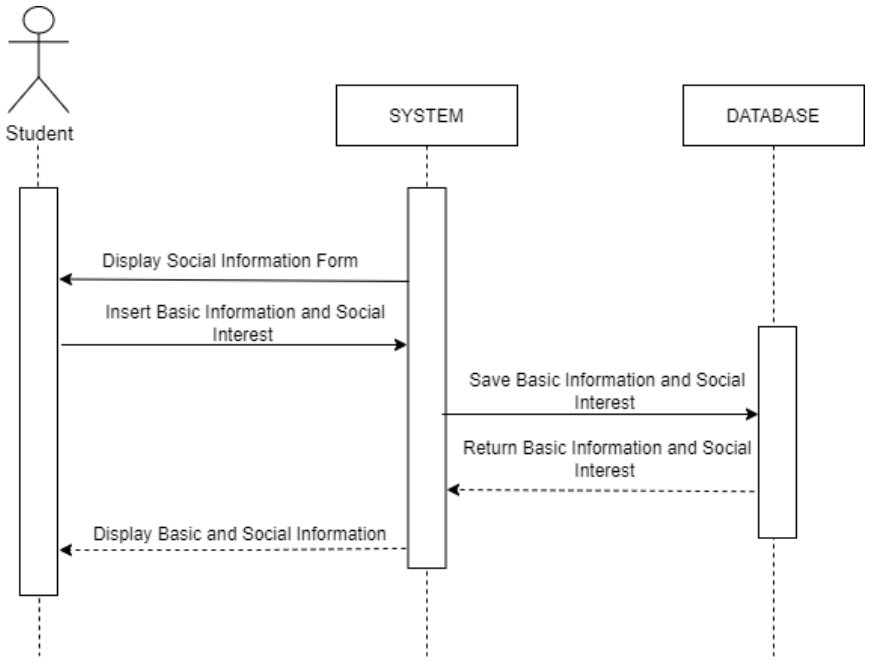


Figure 2.14: Sequence Diagram for Set Basic Information and Social Interest

2.6.2. Use-Case: View and Update Social Profile

2.6.2.1 Identifier: [SRS-009]

2.6.2.2 Use Case Description

Use Case Name:	View and Update Social Profile.
Scenario:	View the student social profile and participation marks for the students club activities and update their basic information and social interest.

Triggering Event:	The student wants to view their social profile and participation marks for their club activities and update their basic information and social interest.	
Brief Description:	As a student, be able to view their social profile and participation marks being displayed at their social profile. and able to update their basic information and social interest.	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Students have filled the information form for their basic information and social interest. Students had joined the respective club events. Students had filled the form to be able to obtain participation marks.	
Postconditions:	Student can view their participation marks being displayed at their social profile page. Students can update their basic information and social interests.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ009-A1] [REQ009-A2] [REQ009-A3] [REQ009-A4]	1. Student view their social profile and their latest participation marks for their club activities. 2. Student click on the "edit" button. 3. The student fills up the edit information form. 4. Student view their latest basic information at their social profile page.	1.1 System fetches the student's latest information and participation marks for the respective club activities from the database. 2.1 System displays the edit information form. 3.1 System saves the input into the database. 4.1 System displays the latest information and participation marks for the respective club activities in the form of charts.
Exception Flow:	The system is offline or inaccessible. The database is offline or inaccessible.	

Table 2.11: Use Case Description for View and Update Social Profile

2.6.2.3 System Sequence Diagram

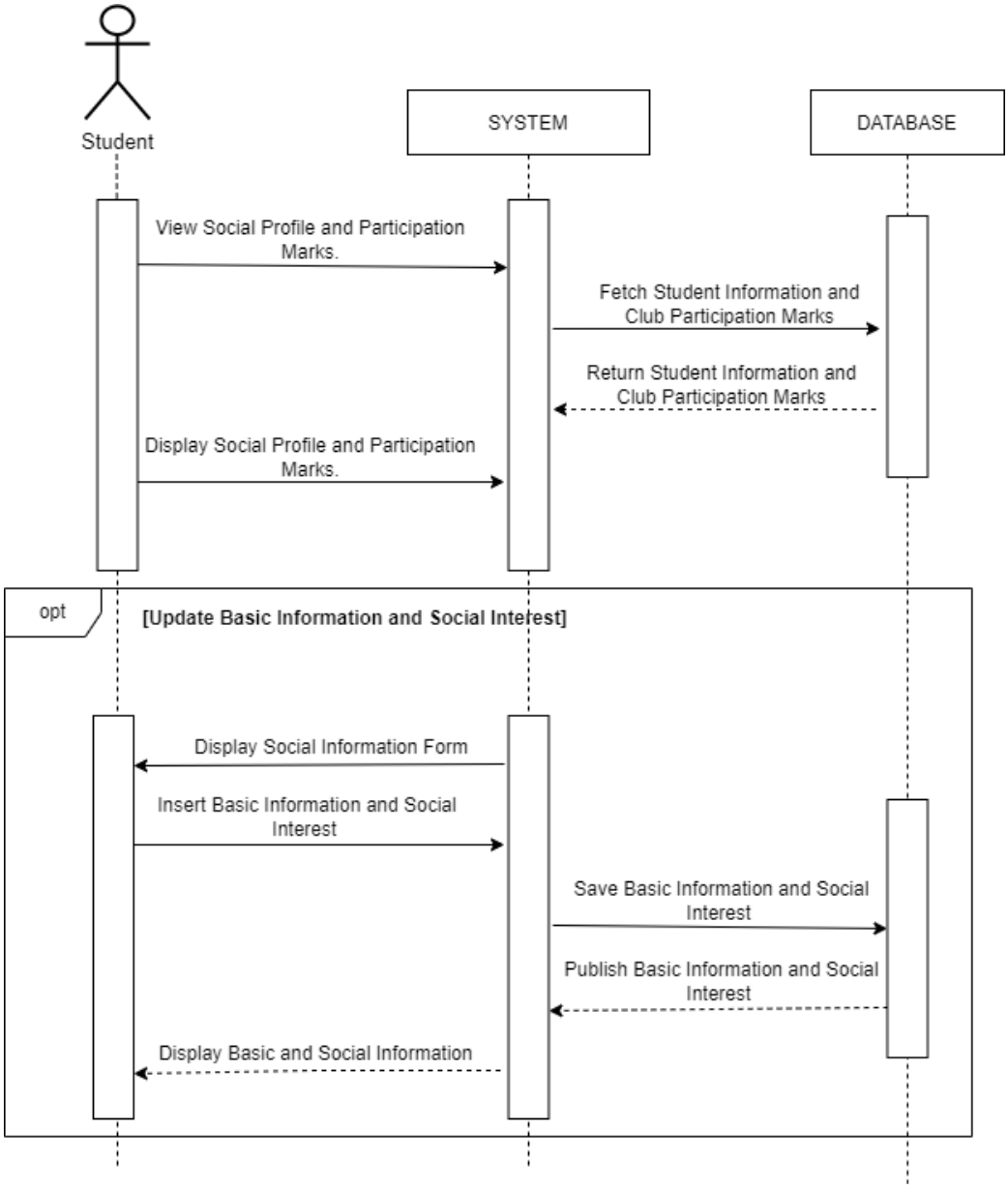


Figure 2.15: Sequence Diagram for View and Update Social Profile

2.6.3. Use-Case: Search Friends

2.6.3.1 Identifier: [SRS-010]

2.6.3.2 Use Case Description

Use Case Name:	Search Friends
Scenario:	View the users of this system and search for students they want to be

	friends with and then is optioned to add them to become friends.	
Triggering Event:	Student want to view and search through the list of users of this system consecutively to add new friends to their social circle.	
Brief Description:	As a student, be able to view a list of other students using this application, search for the friends to add, and option to add them as friends.	
Actors:	Student	
Related Use Cases:	Display Searched Friends Add Friends	
Preconditions:	Students need to be logged into the application. Students had filled up their basic information and social interest in their profile page.	
Postconditions:	Students are able to view other students' profile. Students are able to use the search function to find their friends by inputting their name. Students are able to add their friends into their social cycle.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ010-A1] [REQ010-A2] [REQ010-A3] [REQ010-A4] [REQ010-A5] [REQ010-A6] [REQ010-A7]	1. Student click the “Discover” page. 2. Student view the users of this application. 3. Students search for their friends by inputting their friends’ names. 4. Student successfully receives the results of the searched student and navigates to their social profile. 5. Student view their friend's social profile. 6. Student use the add function to be friends with them. 7. Student successfully add them into their social circle.	1.1 System displays the “Discover” page. 2.1 System fetches the list of students that use this application from the database and displays them under the ‘people’ column in the Discover page. 3.1 System prompts the user to enter the name of the friend they are looking for and sends it to the database. 4.1 The database fetches the name of the student searched and sends it to the system to be displayed. 4.2 System successfully displays the name of the student searched and redirects them to their social profile. 5.1 System fetches the latest information from the database of the user and displays it at their social profile page. 6.1 System saves the add friend information to the database. 7.1 System displays a message to show they had successfully added their friend into their social circle.

Exception Flow:	The database is offline or inaccessible. The system is offline or inaccessible. Unable to be redirected due to internet connection issues.

Table 2.12: Use Case Description for Search Friends

2.6.3.3 System Sequence Diagram

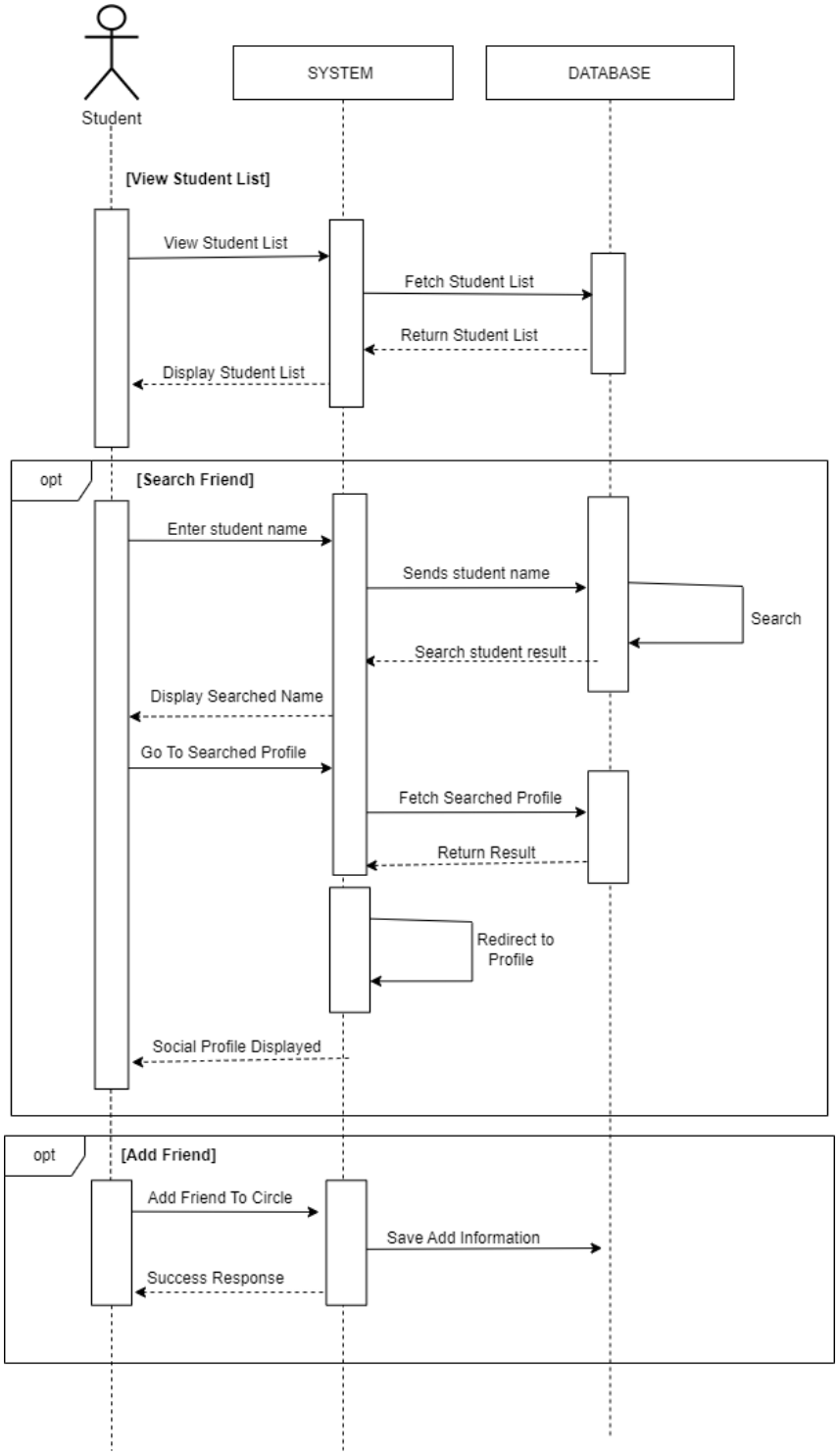


Figure 2.16: Sequence Diagram for Search Friends

2.6.4. Use-Case: Manage Task: View Social Recommendations

2.6.4.1 Identifier: [SRS-011]

2.6.4.2 Use Case Description

Use Case Name:	View Social Recommendations	
Scenario:	View Social Recommendation for the club that is the most suitable for the user based on the interest entered in their social profile.	
Triggering Event:	The student is interested in viewing social recommendations for the club that would be the best fit for them.	
Brief Description:	As a student, be able to view social recommendations based on their interest and can decide to join clubs that align with their interest.	
Actors:	Student	
Related Use Cases:	Join Social Club	
Preconditions:	Student have to be logged into the system. Students had filled up their basic information and social interest in their profile page.	
Postconditions:	A list of clubs that align with the student interest is displayed, and the student is given the option to join any of the listed clubs.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ011-A1] [REQ011-A2] [REQ011-A3] [REQ011-A4] [REQ011-A5]	1. Student click the "Discover" page. 2. Student view the list of recommended clubs. 3. Student click on the club they are interested in joining and navigate to their social page. 4. Student click the join button at the club social page. 5. Student successfully joins the club.	1.1 System displays the "Discover" page. 2.1 System fetches the recommended list of clubs from the database and displays them under the 'Clubs' column in the "Discover" page. 3.1 System redirects the student to the club page. 4.1 System saves the information of the student joining the club into the database. 5.1 System prompts a message that they have successfully joined the club.
Exception Flow:	The database is offline or inaccessible. The system is offline or inaccessible. Unable to be redirected due to internet connection issues.	

Table 2.13: Use Case Description for View Social Recommendations

2.6.4.3 System Sequence Diagram

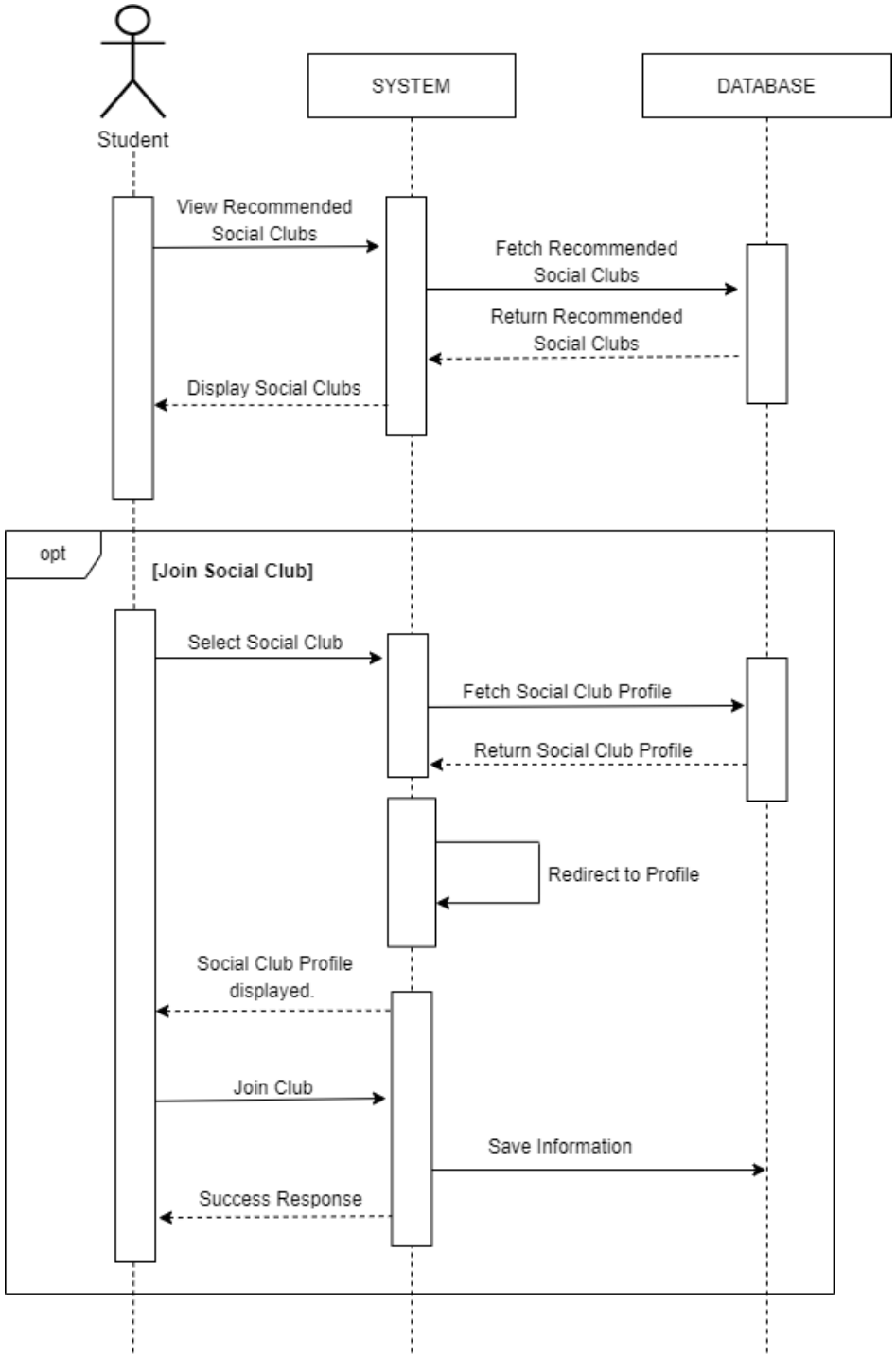


Figure 2.17: Sequence Diagram for View Social Recommendations

2.6.5. Use-Case: View Social Events

2.6.5.1 Identifier: [SRS-012]

2.6.5.2 Use Case Description

Use Case Name:	View Social Events	
Scenario:	View Social Events that are held by the club joined.	
Triggering Event:	The student is interested to see social events that are organized by the club they are a member of and get participation marks.	
Brief Description:	As a student, be able to view social events that are held by the social club they have joined and acquire participation marks.	
Actors:	Student	
Related Use Cases:	Participate in social club events Obtain participation marks.	
Preconditions:	Student have to be logged into the system. Students had filled up their basic information and social interest in their profile page. Student have join the clubs they are interested in.	
Postconditions:	Participate in social events organized by social clubs. Gain participation marks.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ012-A1] [REQ012-A2] [REQ012-A3] [REQ012-A4] [REQ012-A5]	1. Student goes to the social club page. 2. Student click “events” under the social club page. 3. Student click “join” for events they are interested in participating in. 4. Student fill up an information form. 5. Student obtain participation marks from participating in the activity.	1.1 System displays the social club page. 2.1 System fetches the events of clubs from the database and displays them under the “Events” column in the social club page. 3.0 System prompts an information form for the student to fill in. 4.0 System saves data of the student information form into the database. 5.0 System displays the participation marks at the social profile page of the student.
Exception Flow:	The database is offline or inaccessible. The system is offline or inaccessible. Unable to be redirected due to internet connection issues.	

Table 2.14: Use Case Description for View Social Events

2.6.5.3 System Sequence Diagram

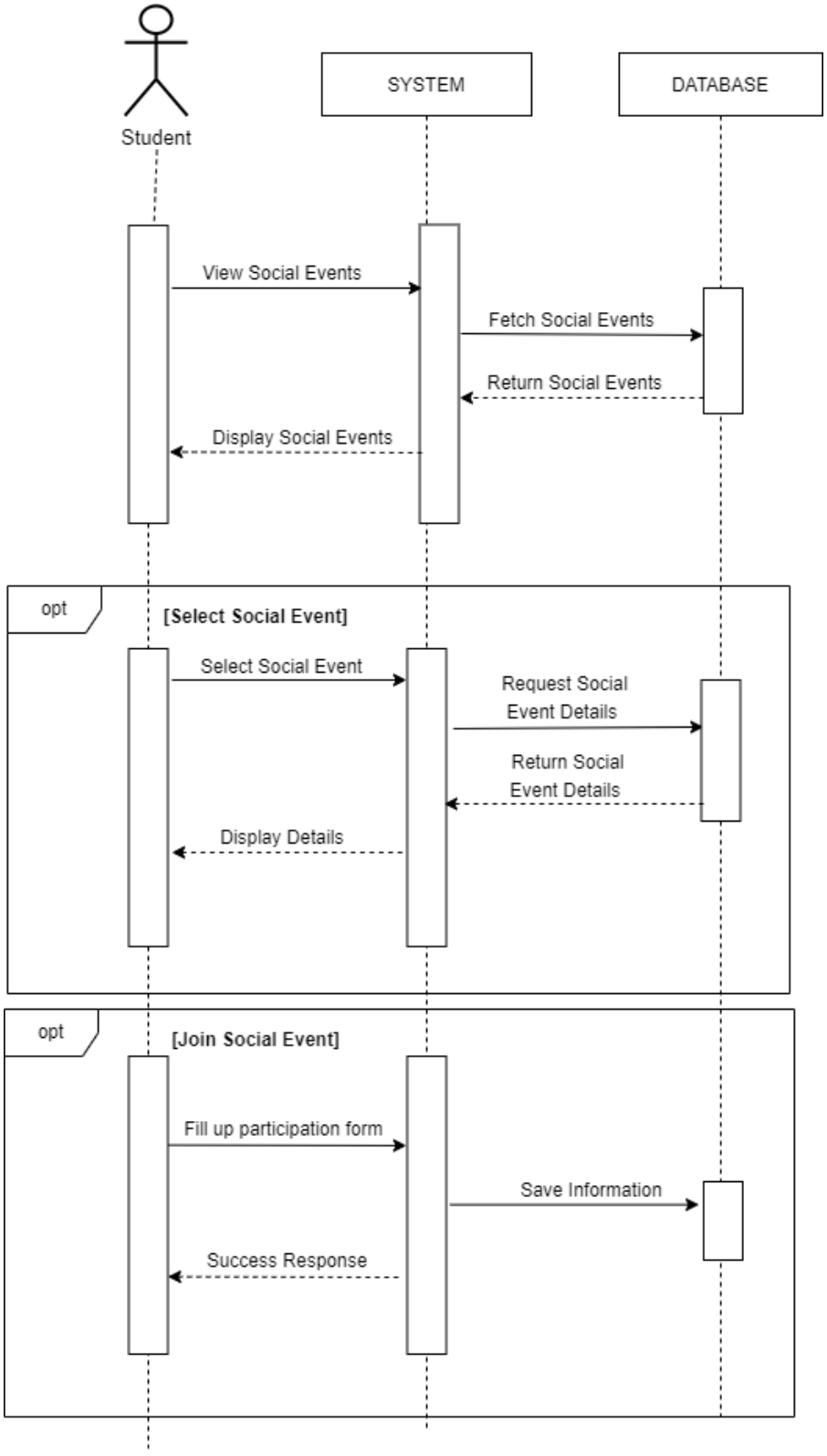


Figure 2.18: Sequence Diagram for View Social Events

2.6.6. Use-Case: Engage with Friends via Chat

2.6.6.1 Identifier: [SRS-013]

2.6.6.2 Use Case Description

Use Case Name:	Engage with Friends via Chat	
Scenario:	Engage using the Chat feature to communicate with friends socially.	
Triggering Event:	Student want to use the Chat feature to engage with their friends.	
Brief Description:	As a student, be able to engage with their friends socially via chat.	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	The student had logged into the system. Student have added friends they want to chat with. Student are at the chat page.	
Postconditions:	Students are able to chat with their friends.	
Normal/Alternate Flow:	**Actor**	**System**
[REQ013-A1] [REQ013-A2] [REQ013-A3]	1. The student goes to the Chat page. 2. Student select friends to chat with by clicking on their profile. 3. Communicate via messaging in the chatbox	1.1 System fetches the list of friends from the database and displays them at the Chat page. 2.1 System redirects them to the chat box when they select the friend they want to chat with and begin chatting. 3.1 System saves the message content into the database.
Exception Flow:	Database is offline or inaccessible. The system is offline or inaccessible. The Chatbox feature is offline or inaccessible.	

Table 2.15: Use Case Description for Engage with Friend via Chat

2.6.6.3 System Sequence Diagram

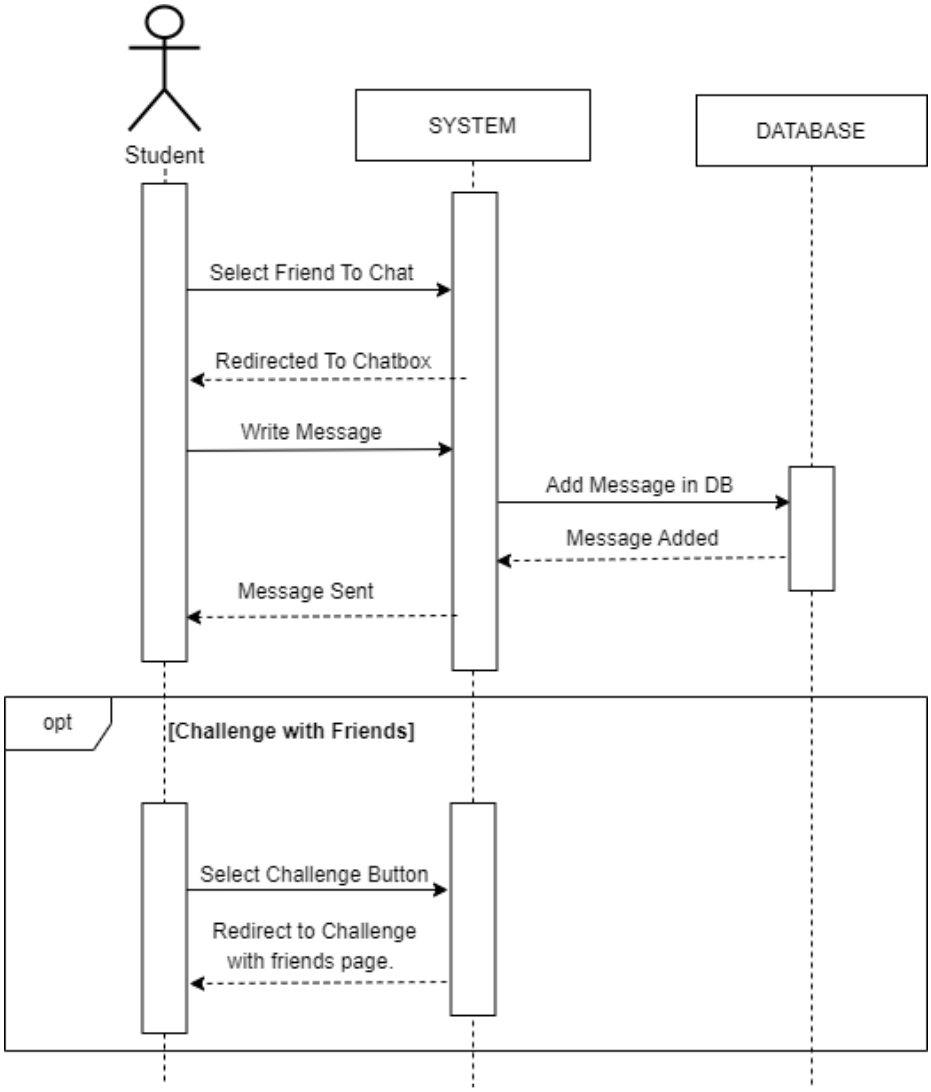


Figure 2.19: Sequence Diagram for Engage with Friends via Chat

2.7. Internal Interfaces Requirements for Fitness Track and Healthy Eating (beFit)

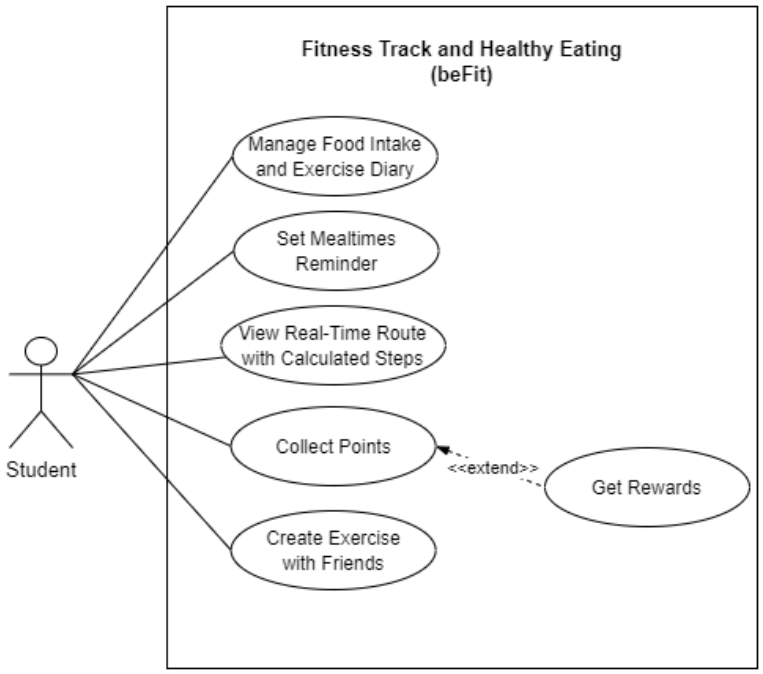


Figure 2.20: Use Case Diagram for beFit Subsystem

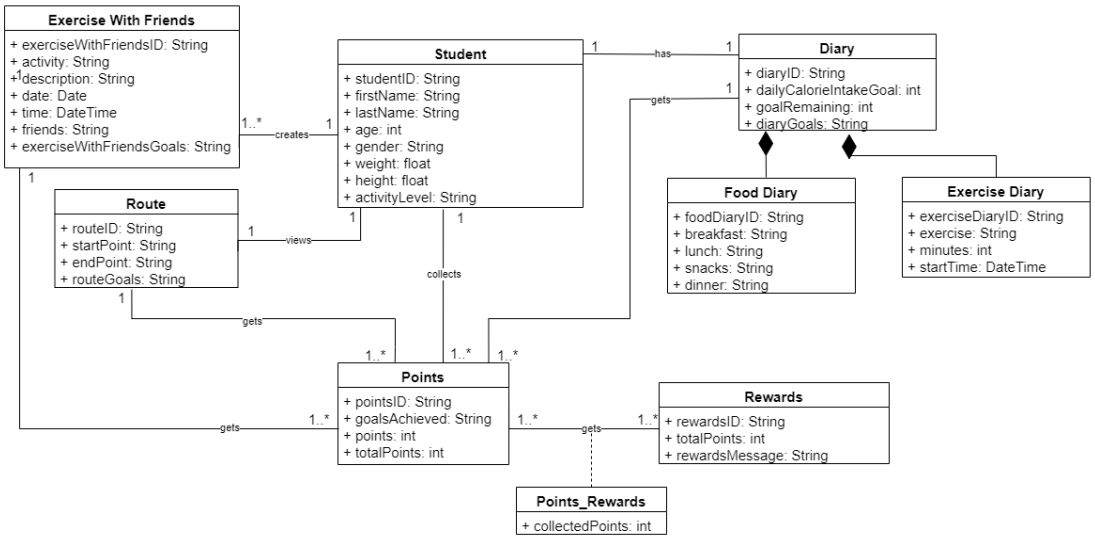


Figure 2.21: Domain Class Diagram for beFit Subsystem

2.7.1. Use-Case: Manage Food Intake and Exercise Diary

2.7.1.1 Identifier: [SRS-014]

2.7.1.2 Use Case Description

Use Case Name:	Manage Food Intake and Exercise Diary	
Scenario:	Manage daily food intake and exercise in the diary	
Triggering Event:	The student wants to manage and track their daily food intake and exercises	
Brief Description:	The student manages the food intake for breakfast, lunch, dinner, snack and exercise done in the diary	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	The student is logged into the system Daily calorie intake goal is calculated	
Postconditions:	- A list of food the student has consumed and exercises they have done will be stored and displayed in the diary - Daily calorie intake goal will be calculated based on the food and exercises the student has done for the day	
Normal/Alternate Flow:	**Actor**	**System**
[REQ014-A1]	1. The student clicks the button “Add Food”	1.1 Direct the page where the student can search the food
[REQ014-A2]	2. The student searches for food	2.1 Display the searched food as well as the calories
[REQ014-A3]	3. The student clicks the food they have searched to add the food	3.1 Store the food in the database and display it in the diary
[REQ014-A4]	4. The student clicks the button “Add Exercise’’	4.1 Direct the page where the student can search the exercise
[REQ014-A5]	5. The student searches for exercise	5.1 Display the searched exercise as well as the intensity level
[REQ014-A6]	6. The student clicks to choose the exercise with the intensity level	6.1 Prompt the student to input the minutes performed and start time
[REQ014-A7]	7. Input the minutes performed and start time	7.1 The system will then calculate calories burned based on the minutes performed
[REQ014-A8]	8. The student clicks save 9. The student updates the food and/or exercise	8.1 Store the exercise in the

[REQ014-A9] [REQ014-A10]	entered 10. The student deletes the food and/or exercise entered	database and display it in the diary 9.1 Update to the database corresponds to the changes made by the student 10.1 Delete the data from the database
Exception Flow: [SRS-014-E9.1] [SRS-014-E10.1]	If the student has not inputted the food and/or exercise first, they cannot update them If the student has not inputted the food and/or exercise first, they cannot delete them	

Table 2.16: Use Case Description for Manage Food Intake and Exercise Diary

2.7.1.3 System Sequence Diagram

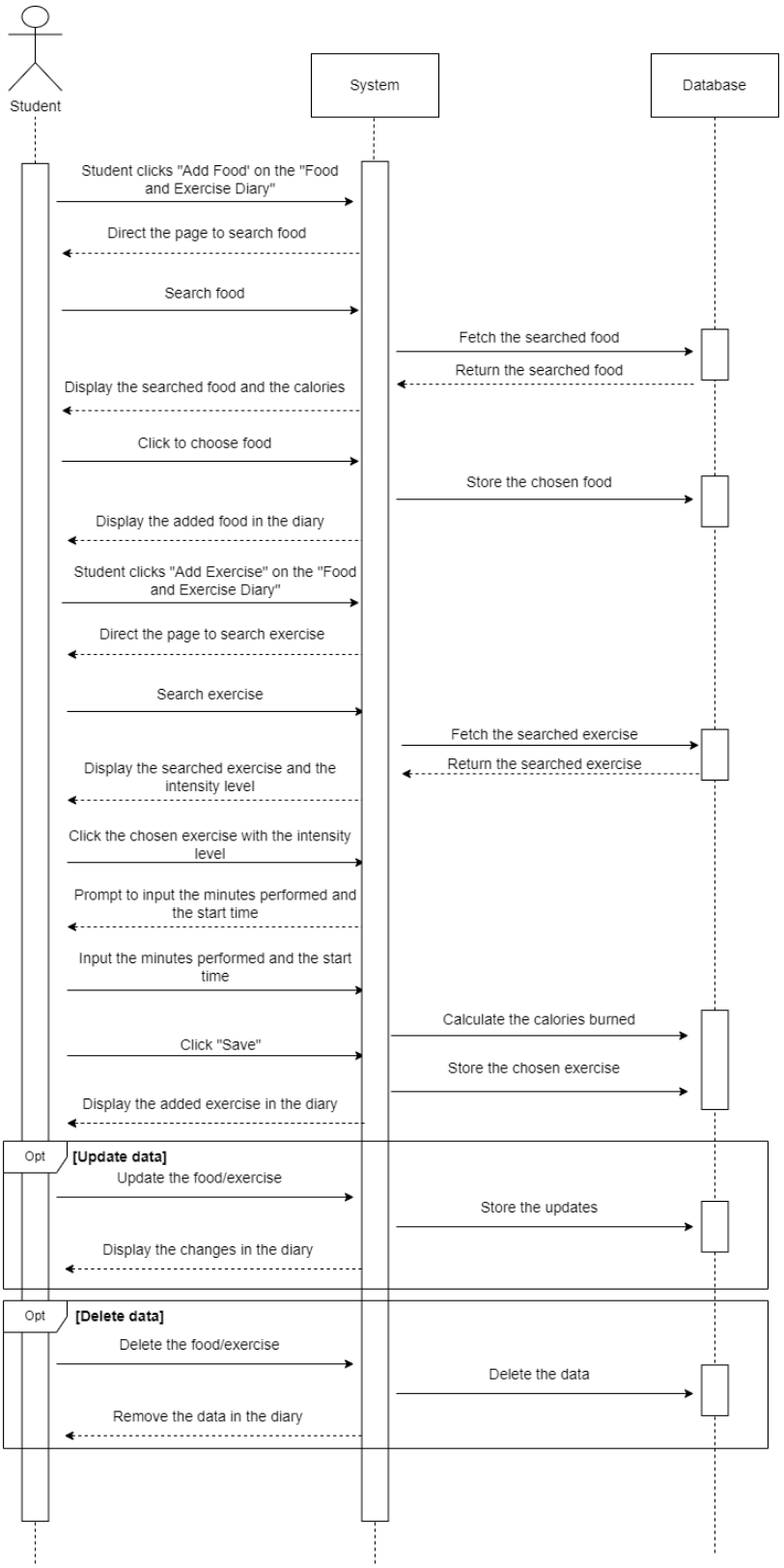


Figure 2.22: Sequence Diagram for Manage Food Intake and Exercise Diary

2.7.2. Use-Case: Set Mealtimes Reminder

2.7.2.1 Identifier: [SRS-015]

2.7.2.2 Use Case Description

Use Case Name:	Set Mealtimes Reminder	
Scenario:	Set reminder for mealtimes	
Triggering Event:	The student wants to be reminded to input in the diary in form of an alarm and to eat each meal	
Brief Description:	Student set reminder for mealtimes such as breakfast, lunch and dinner	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Student is logged into the system	
Postconditions:	Student will be notified on the time that has been set to input in the diary	
Normal/Alternate Flow:	**Actor**	**System**
[REQ015-A1]	1. The student chooses and saves the time for each mealtime; breakfast, lunch and dinner to remind them to input their food intake in the diary	1.1 Store the chosen time to the database 1.2 Alarm the student according to the time set
Exception Flow: [SRS-015-1.1]	If the alarm is turned off, it will not be working	

Table 2.17: Use Case Description for Set Mealtimes Reminder

2.7.2.3 System Sequence Diagram

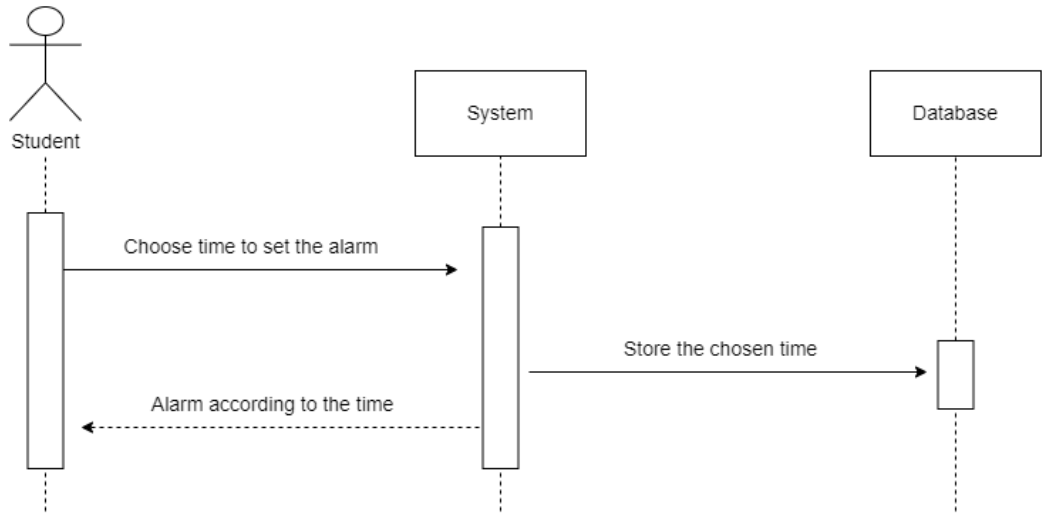


Figure 2.23: Sequence Diagram for Set Mealtimes Reminder

2.7.3. Use-Case: View Real-Time Route with Calculated Steps

2.7.3.1 Identifier: [SRS-016]

2.7.3.2 Use Case Description

Use Case Name:	View Real-Time Route with Calculated Steps	
Scenario:	View the real-time route of the user's steps	
Triggering Event:	The student wants to view their route they have taken with the calculated steps on a map	
Brief Description:	The student views the real-time route of their travelled distance as well as their calculated steps during the walk	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Student is logged into the system Student allows the location to be activated	
Postconditions:	Display the route which the student has travelled along with its calculated steps	
Normal/Alternate Flow:	**Actor**	**System**
[REQ016-A1] [REQ016-A2]	1. Student views the real-time route of their course on a map 2. Student views the steps taken, distance travelled and calories burned from the walking	1.1 Store the route taken by the student 1.2 Display the route 1.3 Calculate the steps taken, distance travelled and calories burned 2. Display the steps taken, distance travelled and calories burned
Exception Flow: [SRS-016-E1.1] [SRS-016-E1.2] [SRS-016-E1.3]	If the location has been turned off, no route will be stored so the student must turn on the location first If the location has been turned off, no route will be displayed so the student must turn on the location first If there is no route stored, the steps taken, distance travelled and calories burned cannot be calculated so the student must turn on the location first	

Table 2.18: Use Case Description for View Real-Time Route with Calculated Steps

2.7.3.3 System Sequence Diagram

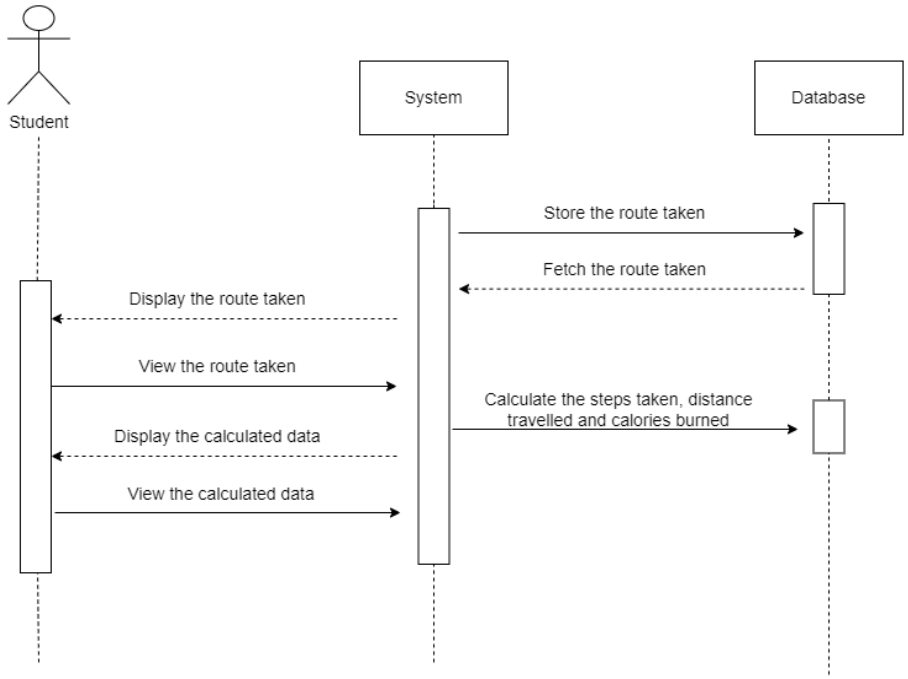


Figure 2.24: Sequence Diagram for View Real-Time Route with Calculated Steps

2.7.4. Use-Case: Collect Points

2.7.4.1 Identifier: [SRS-017]

2.7.4.2 Use Case Description

Use Case Name:	Collect Points	
Scenario:	User of the system collects points after completing goals	
Triggering Event:	The student wants to view their rewards and goals they have achieved	
Brief Description:	After completing any goals, the student will be awarded points that are collected that can be redeemed into reward	
Actors:	Student	
Related Use Cases:	Extend- Get Rewards	
Preconditions:	Student is logged into system Student manages to complete any goals set by the system	
Postconditions:	Student are able to view their collected points to get the rewards	
Normal/Alternate Flow:	**Actor**	**System**
[REQ017-A1]	1. Student views the list of goals they can achieve	1.1 Display a list of goals that can be achieved by the student by completing

		any goals in Food Intake and Exercise Diary, Exercise with Friends and beRoute that is set by the system
[REQ017-A2]	2. The student completes any goal in Food Intake and Exercise Diary, Exercise with Friends or beRoute	2.1 Award the student with points 2.2 Display the total points accumulated by the student 2.3 Display the remaining points to be collected to get rewards
[REQ017-A3]	3. Student views the accumulated points and the remaining points they need to achieve to get the rewards	3.1 Display rewards message to inform the students that they get the rewards
Exception Flow: [SRS-017-E2.1]	The student has not completed any goals	

Table 2.19: Use Case Description for Collect Points

2.7.4.3 System Sequence Diagram

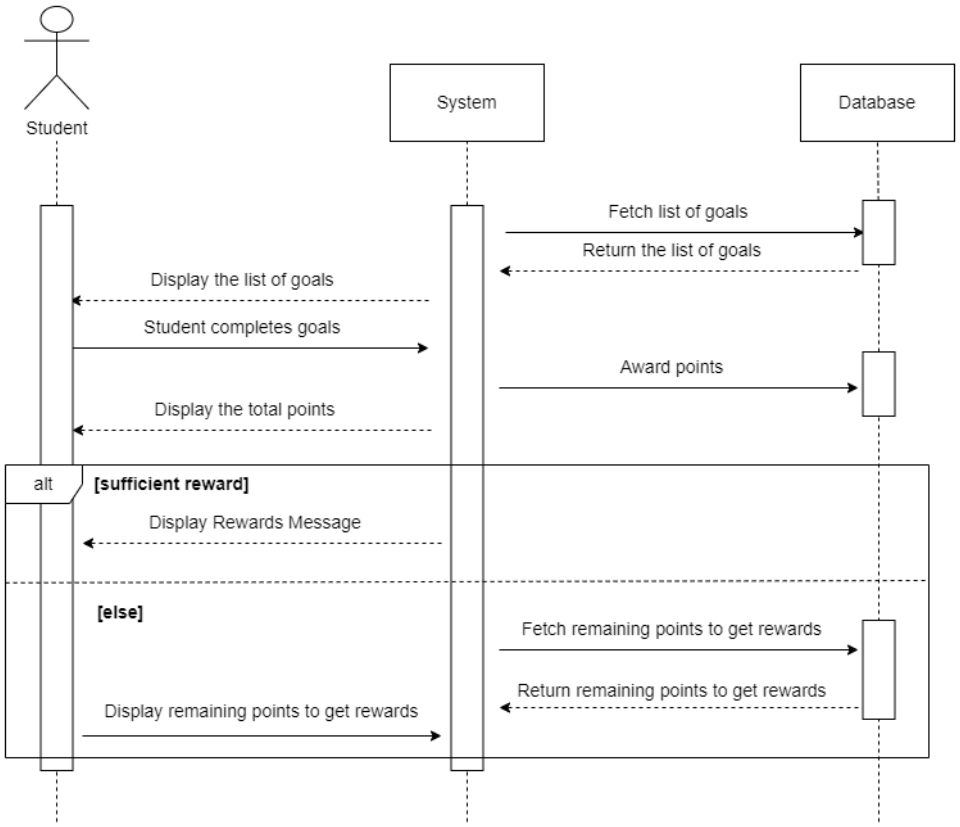


Figure 2.25: Sequence Diagram for Collect Points

2.7.5. Use-Case: Create Exercises with Friends

2.7.5.1 Identifier: [SRS-018]

2.7.5.2 Use Case Description

Use Case Name:	Create Exercises with Friends	
Scenario:	The user of the system creates exercises with friends	
Triggering Event:	The student plans to do exercise with their friends	
Brief Description:	The student is able to create exercises that they plan to do together with friends by inviting them	
Actors:	Student	
Related Use Cases:	-	
Preconditions:	Student is logged into the system Student has at least one group chat Student has a list of friends to be added	
Postconditions:	Student is able to exercise with friends The friends added by the student will be notified	
Normal/Alternate Flow:	**Actor**	**System**
	1. Student fills in the form on the page; title, detail, date, and time. 2. The student adds friends from the list 3. Student clicks the "Save" button	 2.1 Display list of friends from the group chat 3.1 Stores the data 3.2 Display the information in the student's and their friends' task exercise
[REQ018-A2] [REQ018-A3]		
Exception Flow: [SRS-018-E2.1]	Student does not have any group chat Student does not have a list of friends	

Table 2.20: Use Case Description for Create Exercises with Friends

2.7.5.3 System Sequence Diagram

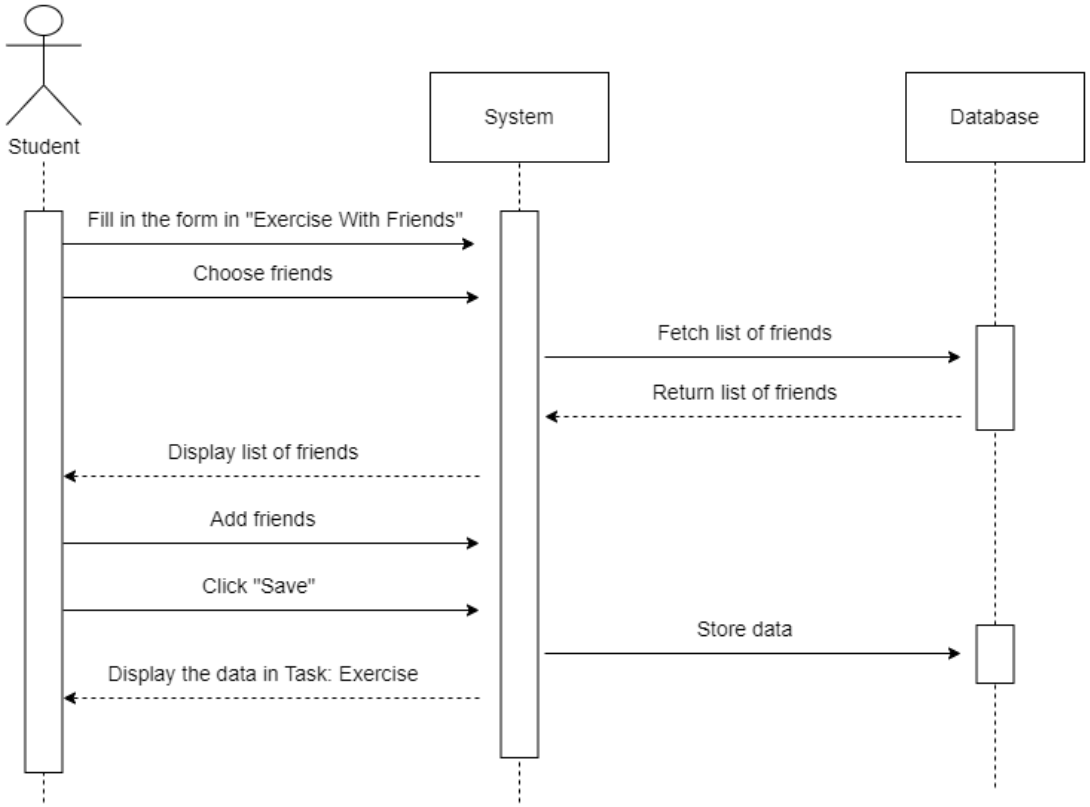


Figure 2.26: Sequence Diagram for Create Exercises with Friends

2.8. Non-Functional Requirements

2.8.1. Space Requirement

A space requirement defines our immediate space needs as well as our future space needs. The system must have enough storage capacity in the storage media to store students' data, recommendation data, and route data for Student Personal Assistance. It also necessitates at least 250GB of free disc space for testing and an additional 150GB for building the code.

2.8.2. Performance Requirement

The following table indicates the system response time limits for processing inputs:

Response Time	Input	Description
Maximum 3	Viewing MyRecord,	• The system must load these pages

seconds	Academic Profile, Social Profile, recommendation courses or social clubs recommendation, and beRoute	and respond to a request from the student. • Example: the system should respond and list out the recommended courses or social clubs' recommendations after the student inserts the data.
Maximum 3 seconds	Inputting any data	• The system should have a response immediately to prompt students for any input on whether to create new data or update data. • Including the time required to process the input, retrieve any necessary data, and return a response to the student.
Maximum 1-second delay	Set alarm or reminder.	• The system should have an alert or remind the student corresponding to the date and time set.
Maximum 3 seconds	Calculate daily calorie intake goal, steps, distance travelled, and calories burned	• The system must calculate when a user sends a request and ends at the time that the application states that the request has been completed

Table 2.21: List of Performance Requirements

2.8.3. Other Relevant Non-Functional Requirement

The final system will have to meet the following performance goals:

Metric	Description	Goals
Usability	Focus on the user interface aesthetics and learnability.	• The user interface of the system is designed to be user-friendly and simple to achieve student satisfaction • Students can use the system without complete guidelines.
Reliability	How likely is the system or its element would run without a failure for a given period of time under predefined conditions	• The system must perform without failure in 95 percent of use cases during a month
Maintainability	Testability and the time required for a solution or its component to be fixed, changed to increase	• All requirements of the system are able to be tested. • The mean time to recovery

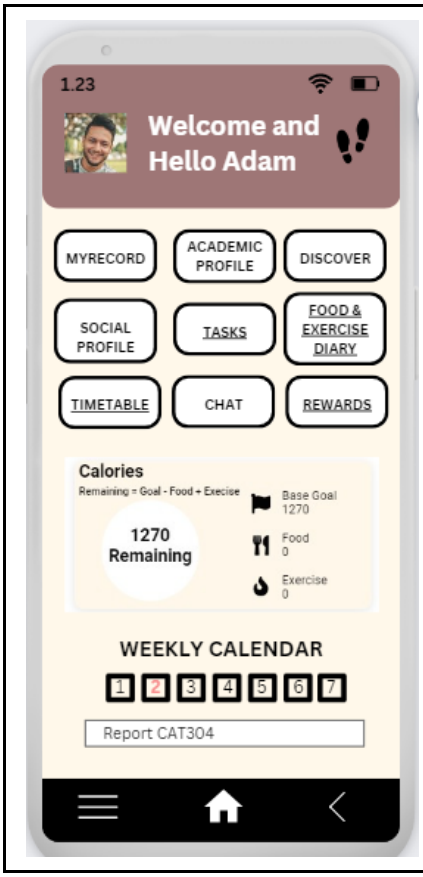
	performance or other qualities, or adapted to a changing environment	(MTTR) following a system failure must not be greater than 10 minutes
Availability	Requirements for the app continuously running	• The system should work 24/7 • The system should function well when it is requested for use.
Security	Data integrity and the data should be secure.	• Student can only access their own account. • The system shall maintain data integrity by keeping backups of all updates to the database

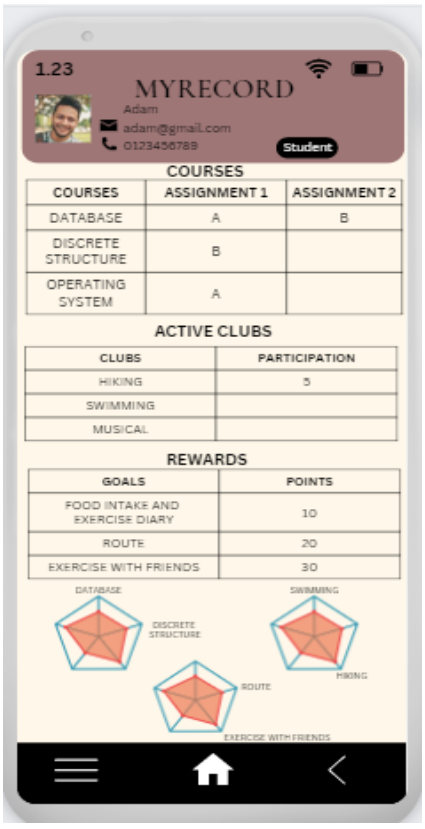
Table 2.22: System Performance Goals

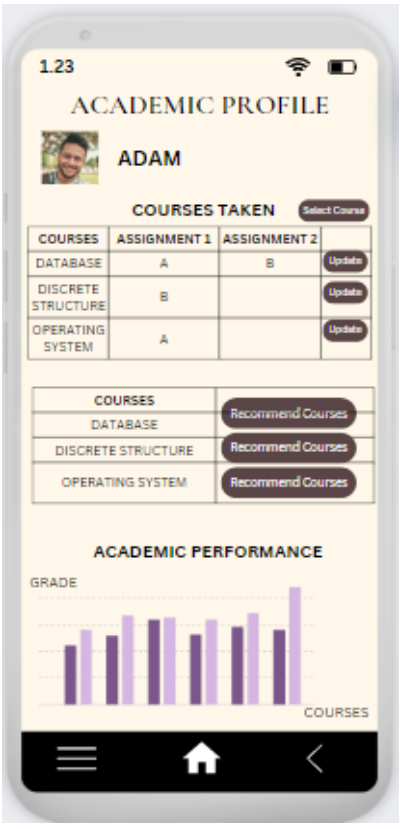
3. Software Design Description (SDD)

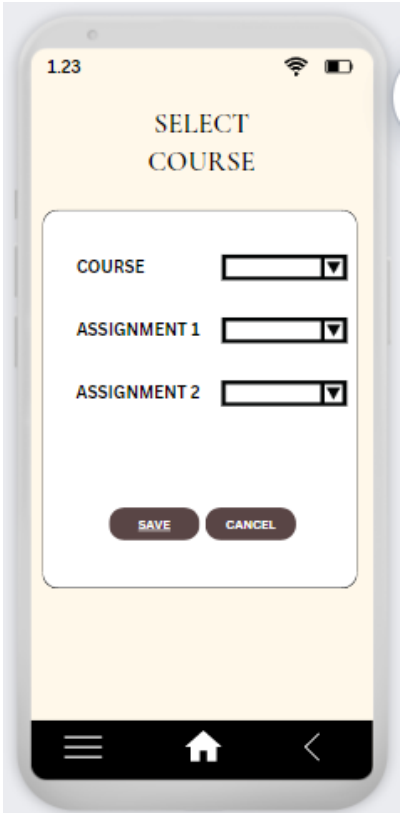
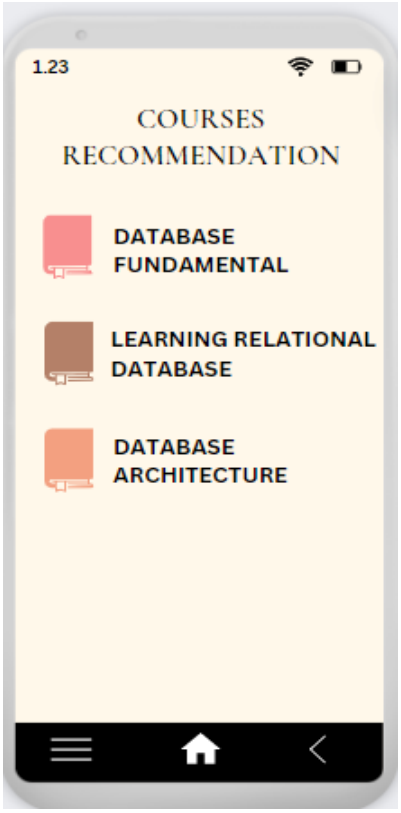
3.1. Storyboard

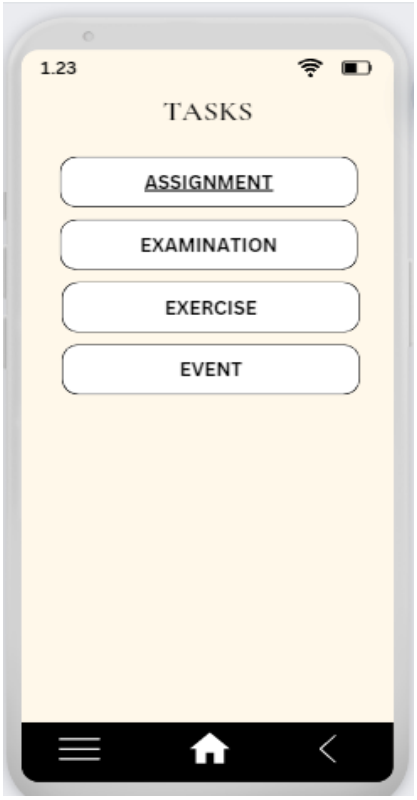
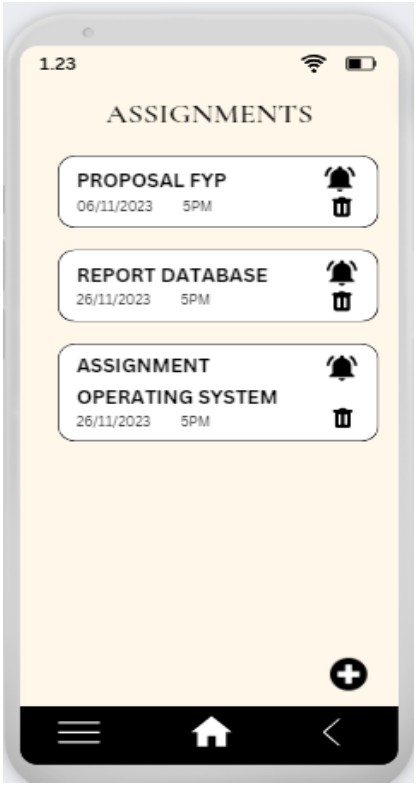
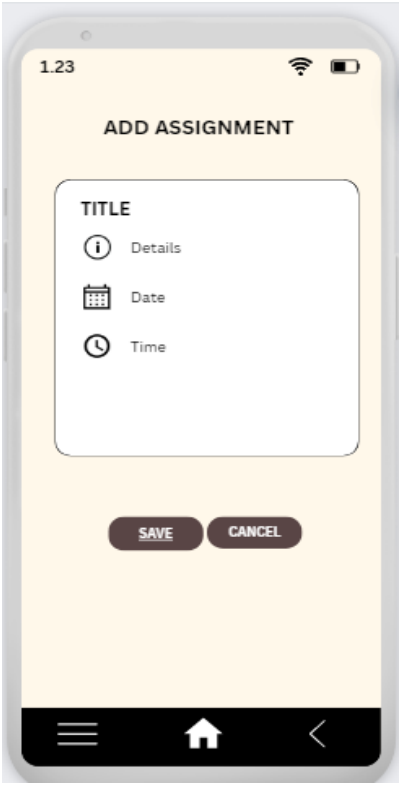
In this section, a storyboard of Student Personal Assistance will be illustrated to understand more of our system interface and flow.

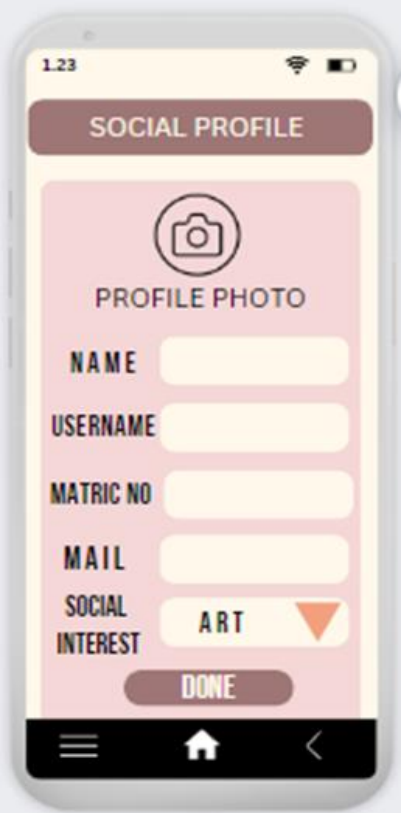
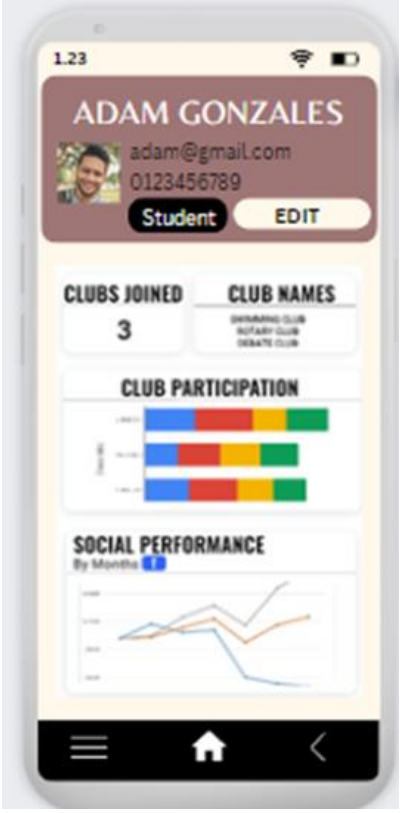


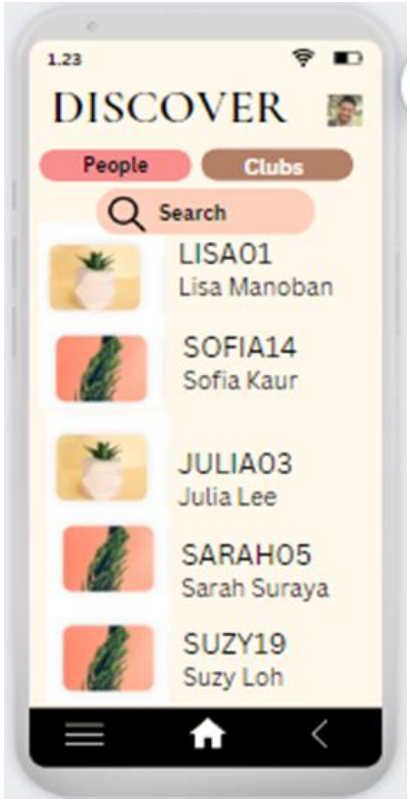
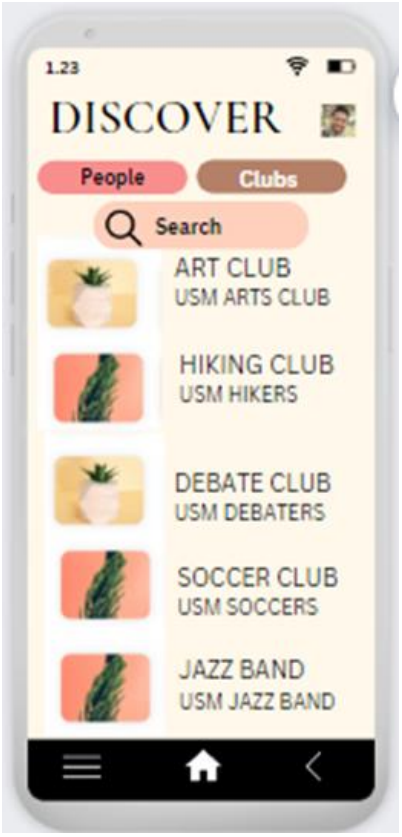


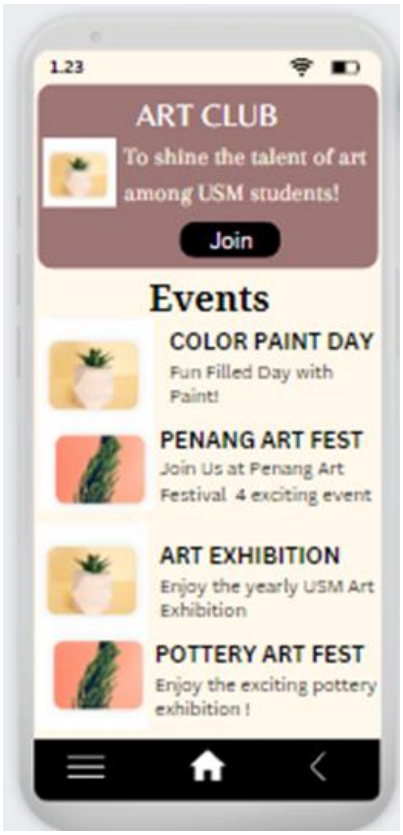
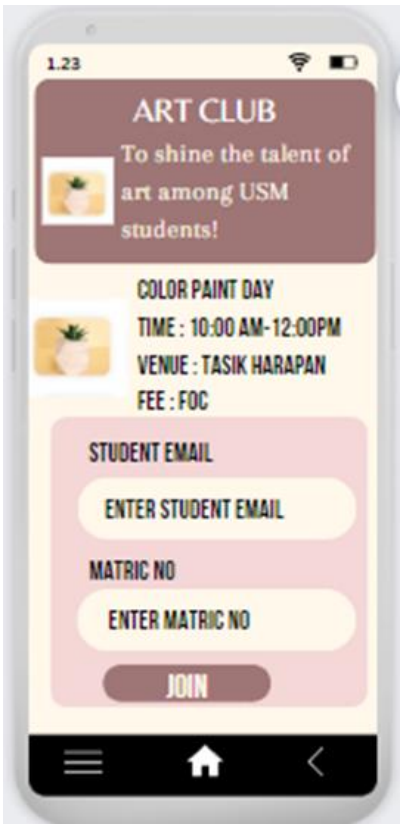


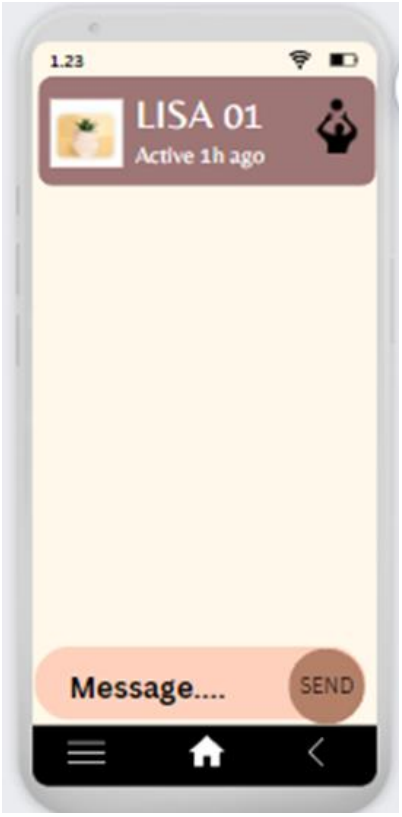
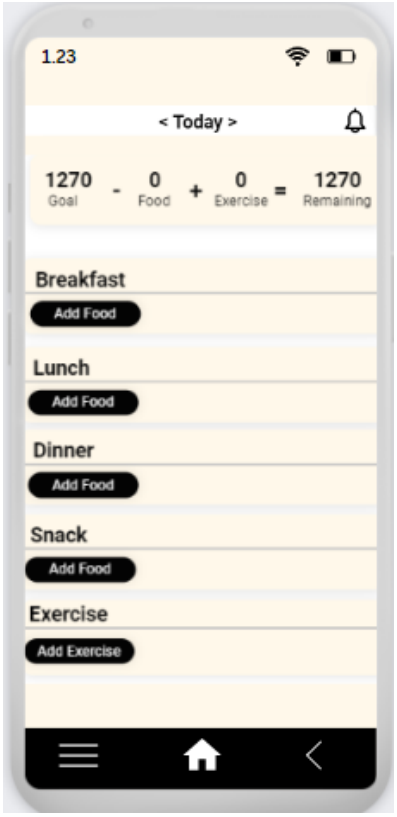
Dashboard interface that consists of navigation, a summary of fitness, and a weekly calendar. The weekly calendar will show the task's dateline for that week.	MyRecord interface that shows student academic, social, and health performance in form of tables and charts.	Interface for Academic Profile that displays Course Taken Section, View Recommend Courses, and Academic Performance in chart form.
		
When the student clicks the "Select Course" button, this interface will be displayed. Prompt the student to select the course to enroll in and insert coursework grades.	This is an interface for students to view courses recommendation after they click the "Recommended Courses" button.	The timetable interface consists of the class list based on the day chosen. Students can click the button "plus" to create a new class. The student needs to click the course name if they want to update and if they want to delete any class they just need to click the "delete" icon.

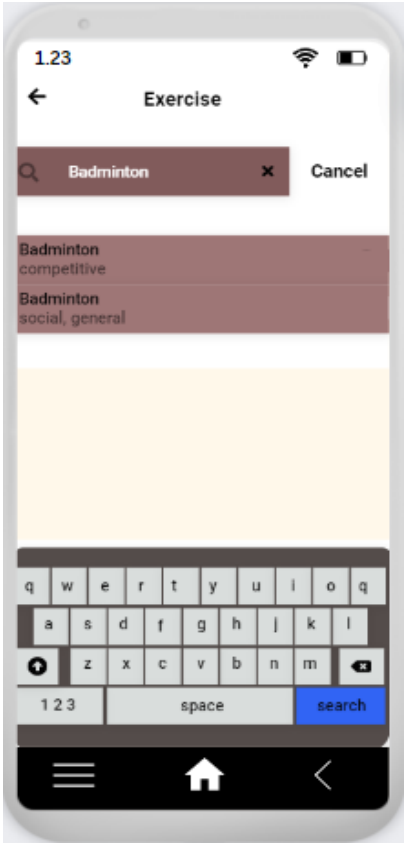
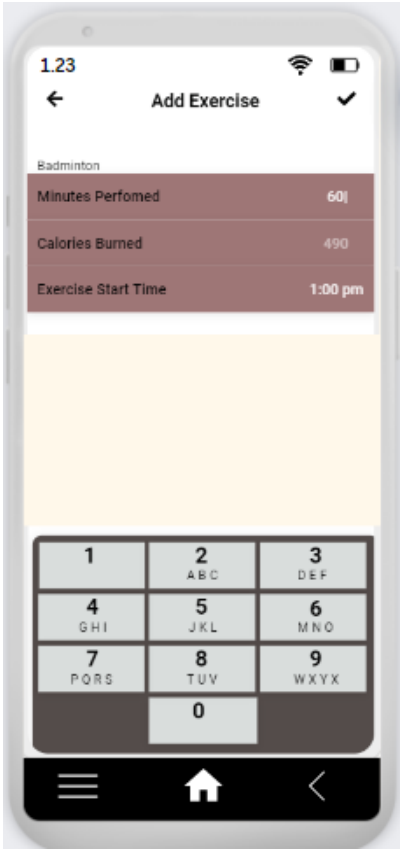
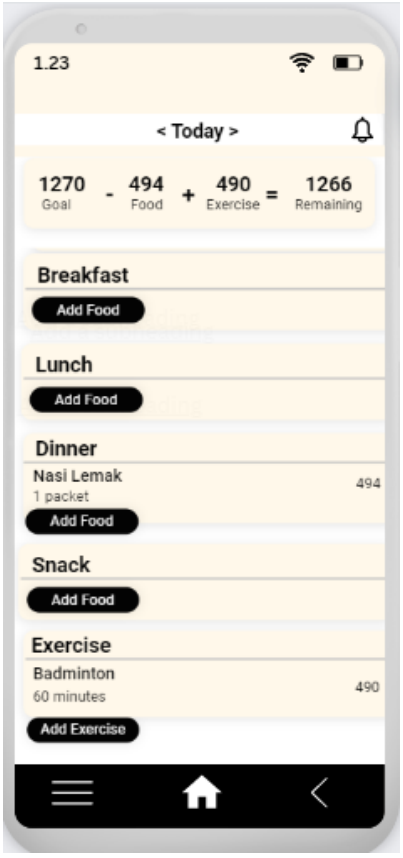
		
If the student chooses "Task" in navigation, the above figure will appear. Task has 4 categories which are the assignment category, examination category, exercise category, and event category.	If the student chooses the assignments category, the list of assignments will display. Students can create new assignments by clicking the "plus" button. The student can click the title in the task to update. If the student wants to delete any task assignment they just need to click the "delete" icon.	Interface for creating a new assignment. This page also allows students to set reminders by setting the date and time of the task. The interface to create a new task exam, task event, and task exercise will be the same as this.

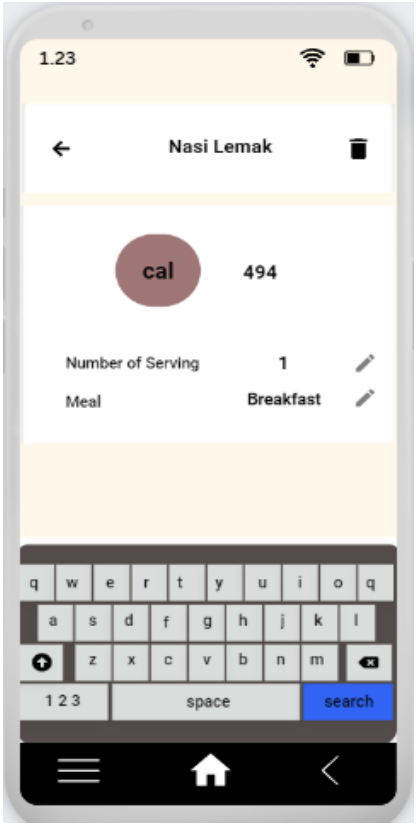
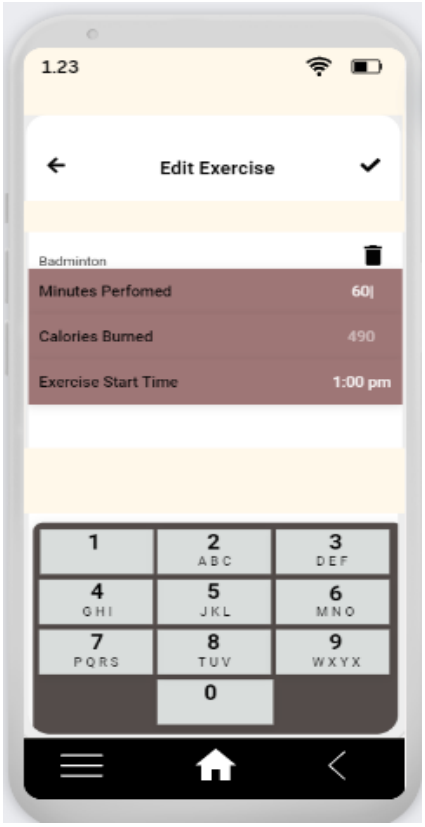
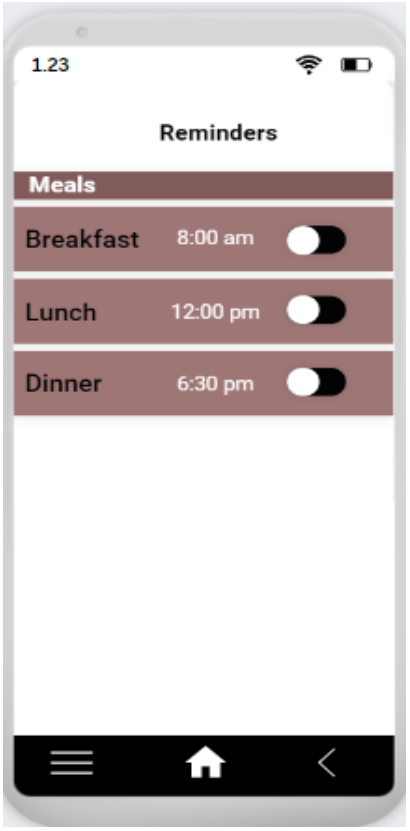
		
Interface for inputting the student's basic information and social interest. The basic information will then be displayed on their respective social profiles and the social interest will be used as an attribute for the recommendation system.	This interface is the student's social profile with their basic information being displayed. It also includes information regarding the student's social activities.	When the student clicks the "edit" button on their social profile, this interface appears for students to update their basic information and social interests.

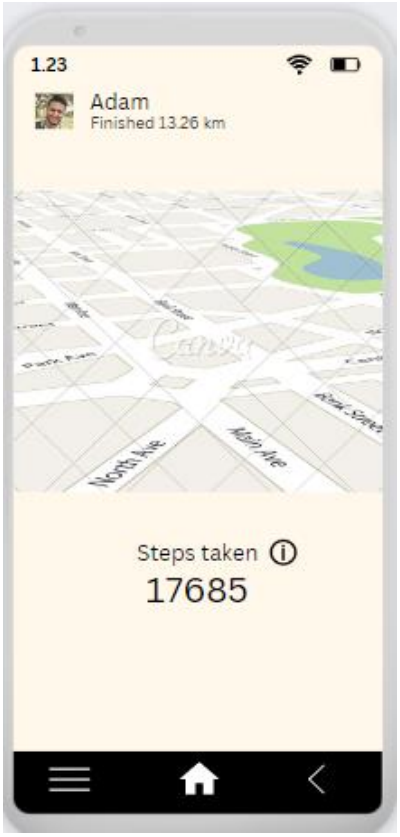
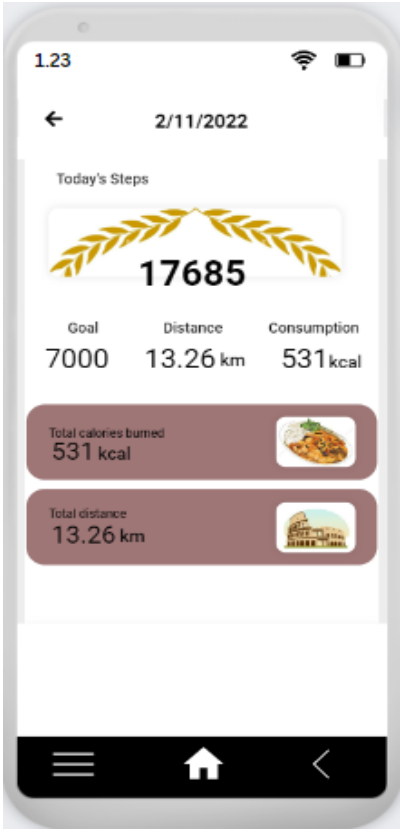
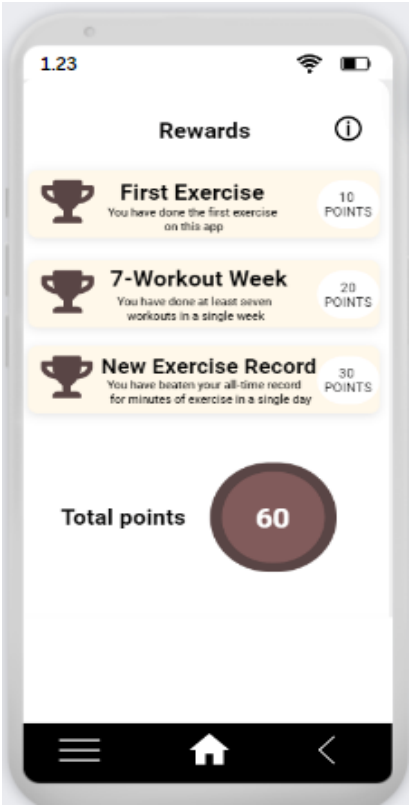
		
Discover interface is divided into 2 categories, people and clubs for students to find friends and social clubs. When the student clicks the ‘People” button, this interface appears showing the list of students that use this application	This interface serves the purpose to enable students to better engage with their peers by adding them into their social circle by clicking the “Add” button.	Discover interface is divided into 2 categories, people and clubs for students to find friends and social clubs. When the student clicks the “Clubs” button, this interface appears showing the list of social clubs that are recommended to the student based on their interest.

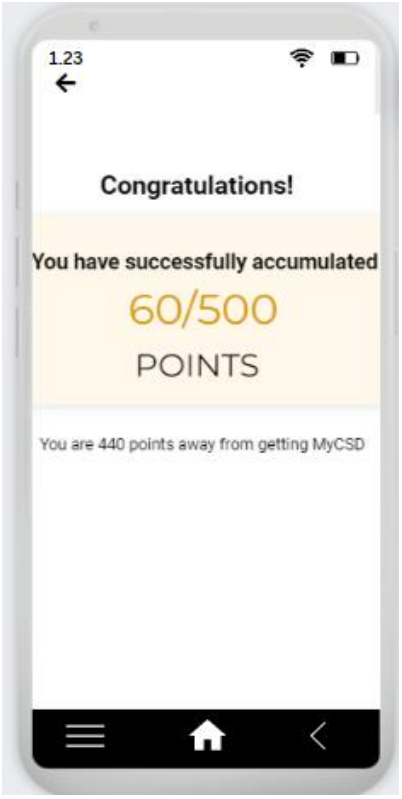
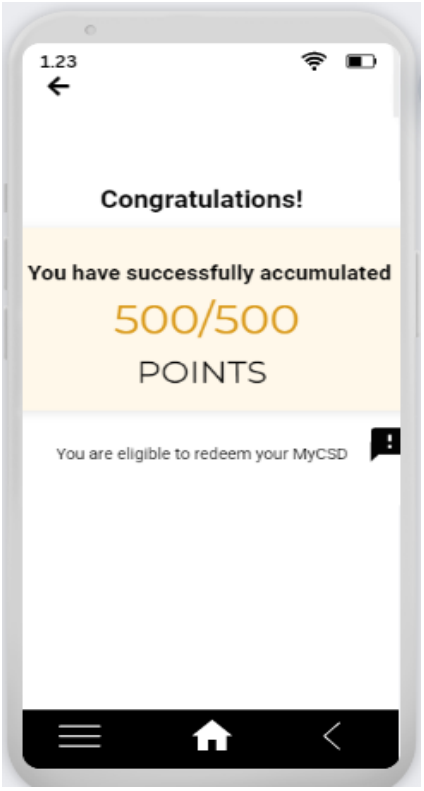
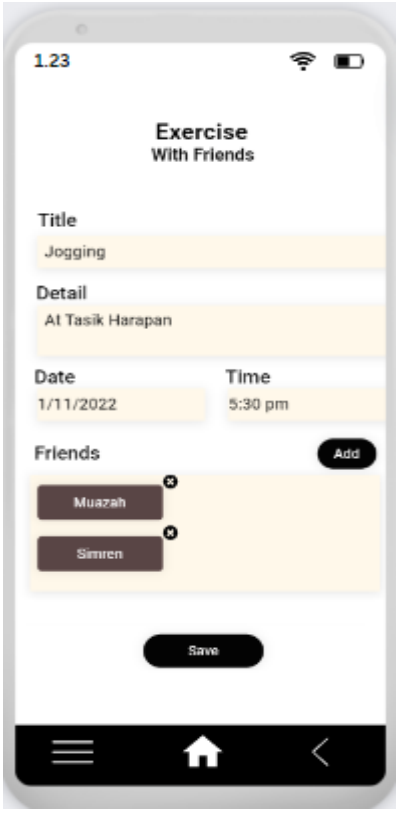
		
When the student clicks on the social club they are interested in, the social profile will be shown. In this, the list of events organized by the social club is shown.	When the student selects the event they are interested in participating in, this interface is shown where the students fill up the participation form in order to get participation marks.	Chat interface serves the purpose of students to better communicate with each other. Here they can search and select for the friend they want to chat with.

		
In this chatbox, the students are able to communicate with each other via message and there is a "Create Exercise With Friends" button, when the button is clicked, the Exercise with Friends interface is shown.	This is the interface for the Food and Exercise Diary that displays the daily calorie intake goal on the top. This diary is also for the students to input their food intake and exercises done.	When the student clicks the "Add Food" in Breakfast, Lunch, Dinner or Snack, this is the interface for them to input the food by searching the food on the search bar. The calorie for the food and the serving will be displayed.

		
When the student clicks “Add Exercise”, this is the interface to search for the exercise and the exercise with its intensity will be displayed.	When the student clicks the exercise with the intensity, they are required to input the minutes performed and the exercise start time. The calories burned will be calculated.	After the student has inputted the food and exercise, it will reflect on the daily calorie intake goal on the top. If food is inputted, it will be subtracted and if exercise is inputted, it will be added. To update or delete, the student needs to click the food or exercise entered.

		
This is the interface to update and delete the food added. It displays the food's calories. When the student wants to update the number of servings or meals, they can click the "edit" icons. To delete the entry, the student can click the "delete" icon.	This is the interface to update and delete the exercise added. The student can update the minutes performed and exercise start time. To delete the entry, the student can click the "delete" icon.	When the student clicks the bell icon in the diary, it will display this interface that has the reminder feature for each mealtime.

		
This is the interface when the student clicks the steps icon on the "Dashboard" that displays the route taken as well as the steps and the distance travelled.	When the student clicks the "information" icon, it will display more details on the steps taken like distance travelled and calories burned from walking.	The interface when the student clicks "Reward" on the "Dashboard" displays a list of the student's achieved goals with their corresponding points and total points.

		
More details on the Rewards when the student clicks the "information" icon that displays the total points and the remaining points to get the rewards	This is the interface when the student manages to collect sufficient point to redeem the rewards.	When the student clicks the "Create Exercise With Friends" button from the chat, this is the interface for them to fill in the details of the exercise

The interface of the exercise that is created from the “Exercise with Friend” feature. After the exercise with friends is completed, it will be added to the “Food Intake and Exercise Diary” under exercise.

Table 3.1: Storyboard

3.1.1. Activity Diagram

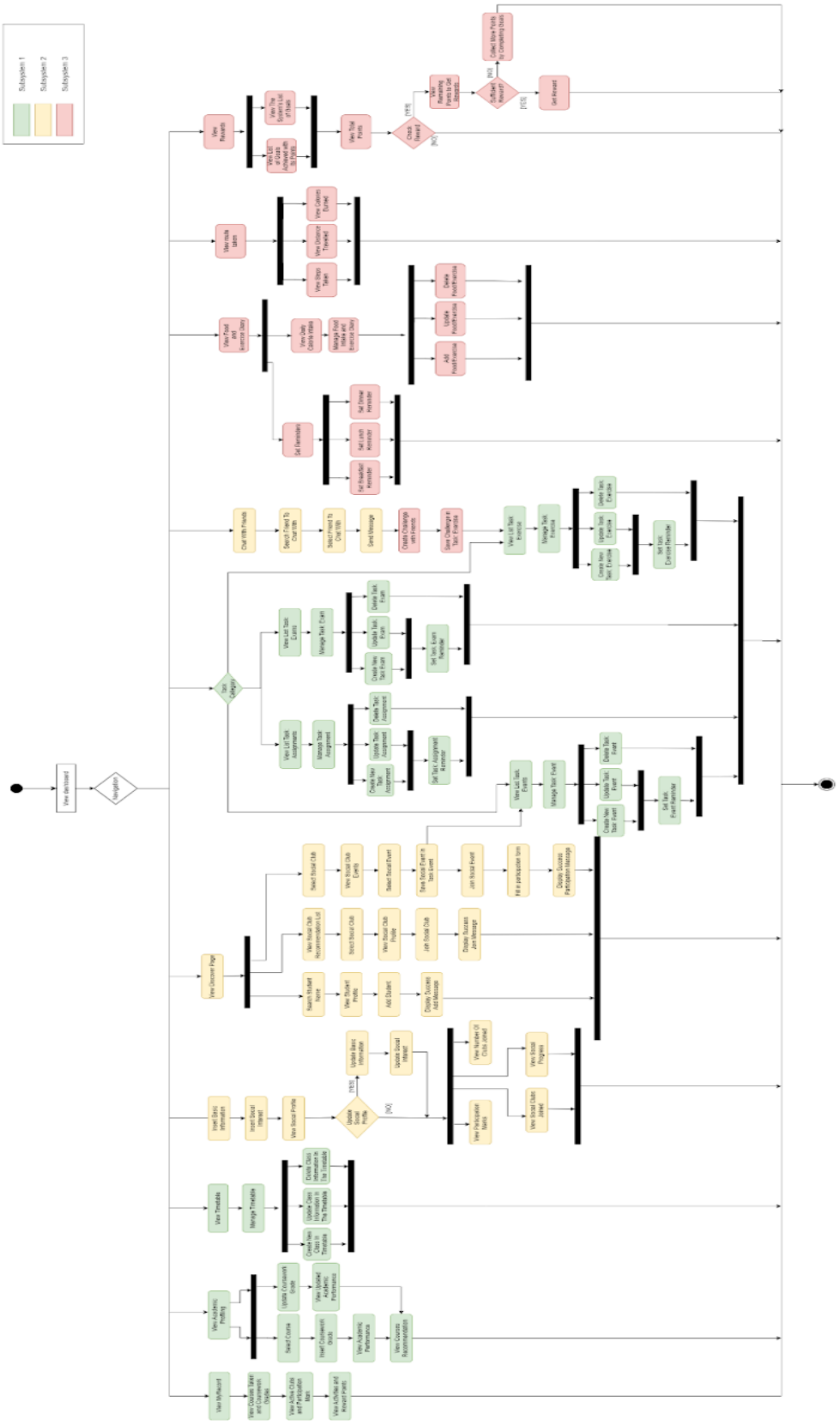


Figure 3.1: Student Personal Assistance Activity Diagram

3.2. High Level Design

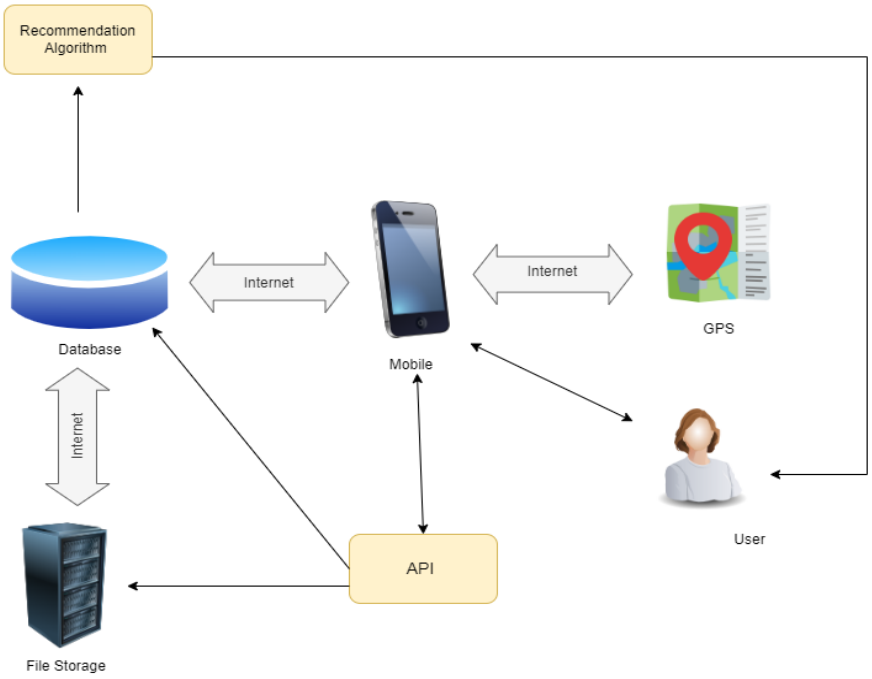


Figure 3.2: *Student Personal Assistance Architecture Diagram*

The system architecture of Student Personal Assistance consists of several key components that work together to provide a seamless experience for our users. As it has been mentioned, it is a mobile app therefore, a mobile app is an integral part of the whole system. The mobile app utilizes the internet to communicate with the database, allowing it to retrieve and display data from the database or save new information to it. Additionally, the database works with file storage to store and manage large amounts of data and files. The file storage, database, and mobile app interact with an API for secure communication and data transfer between the different components of the system. The mobile app utilizes the internet for the GPS component in our system as it will help to accurately track the user's location in real-time. The student uses the recommendation algorithm which then communicates with the database by accessing information from the database, which will be used to recommend the most suitable course and social club for the student based on their selected course and social interests. This information will be saved in the database for future reference. All of these components work together to provide an extensive solution that meets the needs of all of our users.

3.2.1. System Architecture

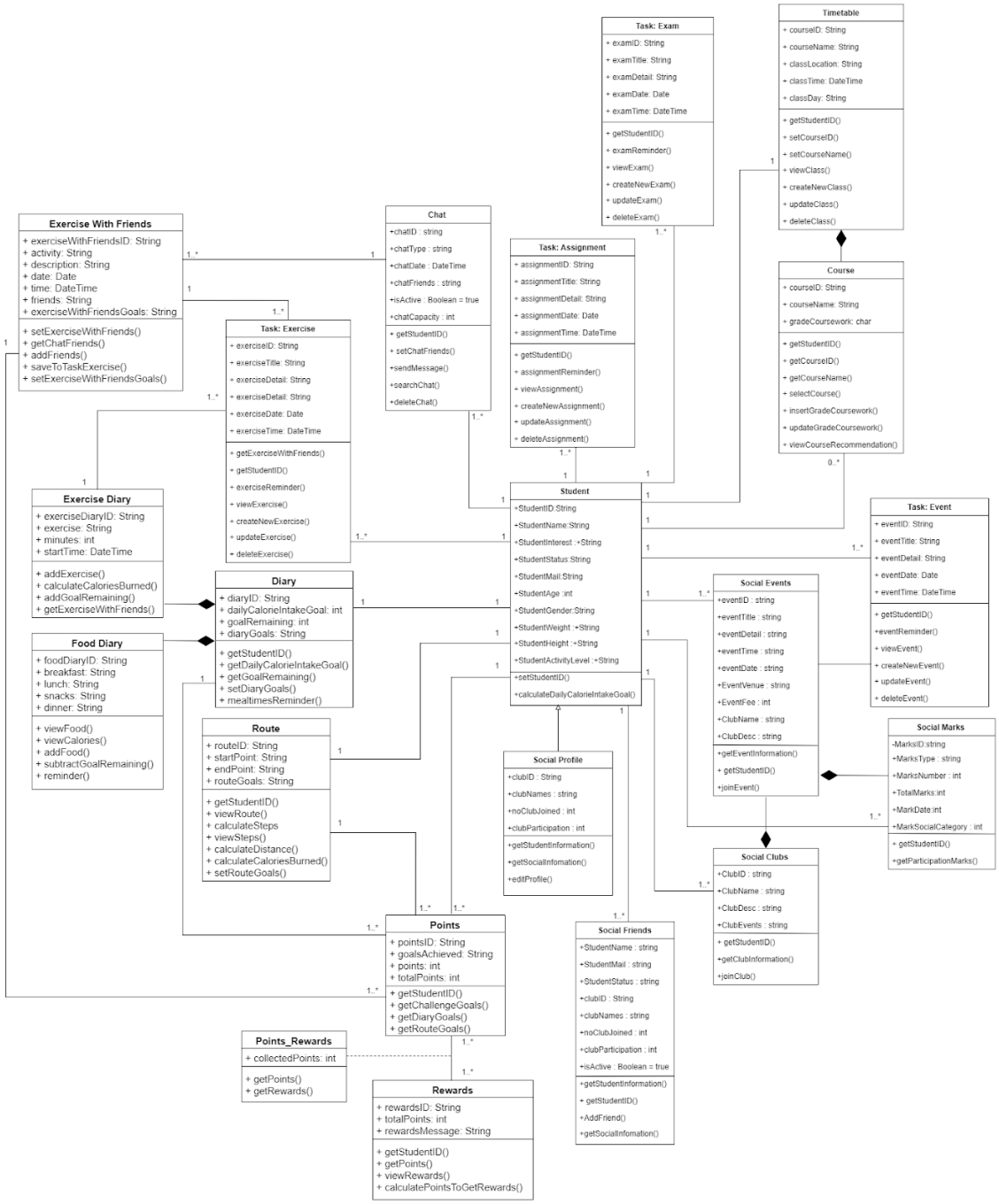


Figure 3.3: Design Class Diagram for Student Personal Assistance

4. Software Test Plan

4.1. Purpose and scope

The Student Personal Assistance system is a comprehensive system that consists of three major subsystems that serve different purposes, each of which contributes to education, socialization, and overall health. Therefore, the scope of testing encompasses all the functionalities offered by these subsystems. Through software testing, we aim to identify any requirements that have not been satisfied in the system. [21] This is crucial for maintaining the quality of the software and for ensuring customer satisfaction with the final product. Discovering and addressing these issues before the software is deployed in critical environments is of the utmost importance.[22] Besides that, one of the main goals of software testing is to identify errors at every stage of development. To achieve this, testing is performed throughout the development process to discover any bugs, issues, or flaws in the code. By identifying these issues early, we can minimize the number of bugs and issues present in the final product, leading to a higher-quality product. [23]

4.2. Test items

The Student Personal Assistance system is being tested as a whole, and all components within it are test items. Student Personal Assistance is a system designed to help students manage their lives in university by providing them with resources for academic, health, and social needs. For academic wise, iStudy is developed where it focuses on organizing the schedule and monitoring their academic life by allowing the student to view academic performance and recommended courses. The next subsystem covers the social aspect of a student's life which is MyCircle. This subsystem focuses on enhancing the social aspect of student life by providing a platform for students to connect with each other and participate in social clubs that are recommended to them and promoting communication and engagement among students through the chat feature available. The third subsystem is beFit which focuses on the healthy life of the students in terms of fitness tracking and healthy eating to keep adopting a healthy lifestyle.

4.3. Requirements/Features to be tested

Identifier	Requirements / Features	Related Requirements / Features	Description
Subsystem 1: My Study Life (iStudy)			
REQ-001	View MyRecord	-	This feature allows students to view their academic, social, and health performance in both table and chart form.
REQ-002	Select courses	- Insert grade coursework - View courses recommendation	This feature allows the student to select a new task and insert its coursework grades. The student is also able to view course recommendations. The system also allows student to update their coursework grades.
REQ-003	Manage timetable	-	The student can create a new class, update class information as well as delete a class. Students should be able to view their timetable after they create or update or delete.
REQ-004	Manage task: Assignment	Set task: Assignment reminder	The student can use the basic features such as view, create, update and delete. The other feature is students are also allowed to set and receive reminders.
REQ-005	Manage task: Exam	Set task: Exam reminder	The student can use the basic features such as view, create, update and delete. The other feature is students are also allowed to set and receive reminders.
REQ-006	Manage task: Event	Set task: Event reminder	The student allows to use the basic features such as view, create, update and delete. The other feature is students are also allowed to set and receive reminders.
REQ-007	Manage task: Exercise	Set task: Exercise reminder	The student can use the basic features such as view, create, update and delete. The other feature is students are also allowed to set and receive reminders.
Subsystem 2: My Social Network (MyCircle)			
REQ-008	Set basic information and	-	This feature is for the student to set their basic information such as inserting their

	social interest.		name, ID, and student email to be displayed at their social profile and inserting their social interest for the system to recommend suitable social clubs for the student to join.
REQ-009	View and Update Social Profile	-	This feature is for students to view their social profile which includes information about their participation in social clubs, marks earned from social activities, and overall level of social engagement. This feature also enables students to keep their personal information and social interest current.
REQ-010	Search Friend	- Display Searched Friends. - Add Friends	This feature is for the student to view the list of students using this app and search for specific friends by name. The result of the search will be displayed and it will redirect the user to the searched social profile where the students are able to use the 'Add' feature to add them to become friends and include them to their social circle.
REQ-011	View Social Recommendation	- Join Social Club	This feature allows student to view a list of social clubs that match their interests, as specified in their social profile. By clicking on a particular social club they are interested in, it will redirect them to the social club's profile where they can use the 'Join' feature to join the social club.
REQ-012	View Social Events.	- Participate in Social Club Events. - Obtain Participation Marks.	This feature is for the student to participate in social club events. By selecting the event they are interested in, they can set a reminder for the event. On that day the student is able to fill up the participation form to earn participation marks from it.
REQ-013	Engage with Friends via Chat	-	This feature is for the student to use the chat feature available. Students are able to select friends they want to have a chat with and it will redirect them to the chat box where they are able to send messages to each other. There is also an "Exercise with Friends" button where it redirects them to the 'Exercise with

			Friends' page.
Subsystem 3: Fitness Track and Healthy Eating (beFit)			
REQ-014	Manage Food Intake and Exercise Diary	-	This feature has a daily calorie intake goal that includes remaining, goals, food, and exercise. The student can add, view, update and delete the daily food intake and exercise done in the diary. When food intake is added, the daily calorie intake goal is subtracted and when exercise is added, the daily calorie is added
REQ-015	Set Mealtimes Reminder	-	This feature is for the student to set a reminder for mealtimes; breakfast, lunch, and dinner
REQ-016	View Real-Time Route with Calculated Steps	-	This feature allows the student to view their route taken that also track steps then calculate the steps, distance travelled and calories burned from the route
REQ-017	Collect Points	Get Rewards	This feature is for the student to collect points after completing goals set by the system and then get the rewards after collecting sufficient points
REQ-018	Create Exercise with Friends	-	This feature allows the student to create exercise with friends that they plan to do together. It also enables the student to invite friends from a group chat that lists the friends possible to be added

Table 4.1: List the features to be tested

4.4. Requirements/Features not to be tested

Requirements / Features	Justification
Display Searched Friends.	This feature does not require individual testing, as it will be thoroughly tested as part of the Search Friend feature. When the Search Friend feature is tested, the display of search results will be evaluated as a pass or fail criteria. Conducting a separate test specifically for the Display Search Friend feature would be unnecessary and inefficient.

Table 4.2: List the features not to be tested

4.5. Test approach/strategy

Unit testing is the first round of the testing phase, it refers to evaluations of the system's each components, such as its functions and processes, to make sure that each one operates correctly on its own. White box testing technique will be used to assess the software's code and internal structure. The advantage of using this testing phase is that every time a piece of code is modified or altered, it allows for rapid problem-solving.

Besides, integration testing also will be used during the implementation of the Student Personal Assistance system. Integration testing is the phase where each unit component combines within a system and is tested in a group. Problems with how many system functions interact with one another can be identified and found using this testing stage. Through integration testing, the overall effectiveness of units when they operate together can be determined [24]. This stage is critical since the program's overall functionality depends on the units operating continuously as a whole system and not as separate processes.

The system testing phase is crucial to implement as the system is put through its first test as a fully integrated application. The aim of using system testing is to determine how effectively it serves its intended purpose and to ensure that the system complies with all the requirements specified and to ensure that it satisfies quality standards. System testing is essential since it makes sure that the system complies with the functional, technical, and business requirements that were set by the client.

Acceptance Testing is the final stage of software testing for our project that occurs after System Testing and before the system is made available for use. It is formal testing based on user needs, requirements, and business processes to determine whether a system meets the acceptance criteria or not and to allow users, customers, or other authorised entities to decide whether or not to accept the system [25]. In conclusion, these four stages of testing will be used during the development of our system, Student Personal Assistance.

4.6. Item pass/fail criteria

Identifier	Requirements/ Features	Related Requirements	Pass Criteria	Fail Criteria
Subsystem 1: My Study Life (iStudy)				
REQ-001	View MyRecord	-	- Able to click "MyRecord" and display "MyRecord" page detailing academic, social, and health performance. - Display the correct information	- When clicking "MyRecord" it displays a wrong page or does not display anything (blank page). - Display wrong information.
REQ-002	Select courses	- Insert grade coursework - View courses recommendation	- Able to click the "Select Courses" button, then select the course and insert its coursework grades. - Able to click "Recommend Courses" and it displays recommended courses list. - Able to click the "Update" button, then update the coursework grades and it displays the changes made. - Display the academic performance chart after student inserts or update their coursework grades.	- Display the wrong page or/and does not allow the student to select the course and insert its coursework grades when clicking the "Select Courses" button. - When clicking the "Recommend Courses" button, it displays the wrong page. - Does not allow the student to update their coursework grade or/and did not display the updates made. - Display the wrong academic performance chart after student inserts or update their coursework grades.
REQ-003	Manage timetable	-	- Students are redirected to the	- When clicking "Timetable" it

			timetable page when clicking the "Timetable" button. - Able to create a new class by clicking the "plus" button and the class created is displayed. - Able to click the course name, then update class information and it will display changes made. - Able to delete any class by clicking the "Delete" icon and it will be deleted from the timetable.	redirects the student to the wrong page. - Does not allow the student to create a new class or/and display the wrong page when the student clicks the "plus" button. - Does not allow the student to update class information or/and did not display the changes made. - Unable to delete the class or/and the class deleted is not removed from the timetable.
REQ-004	Manage task: Assignment	Set task: Assignment reminder	- A list of task assignments is displayed when the student clicks the "Assignment" button. - Able to click the "plus" button to create a new task assignment and the new task assignment created will be displayed. - Able to set a reminder and receive reminders for each task assignment on time. - Able to click the title of the task assignment to update task	- Redirected student to the wrong page when clicking the "Assignment" button. - When clicking the "plus" button, students can not input the task assignment information, or/and the new task assignment created is not displayed. - Unable to set the reminder for each task assignment or/and receive reminders before or later than the time and date set or do not receive reminders at all.

			assignment information and it will display changes made. - Able to click the "Delete" icon to delete any task assignment and it will be removed from the list of task assignment.	- Does not allow the student to update task assignment information or/and changes made did not display. - Unable to delete the task assignment or/and the task assignment deleted is not removed from the list of task assignment.
REQ-005	Manage task: Exam	Set task: Exam reminder	- A list of task exams is displayed when the student clicks the "Exam" button. - Able to click the "plus" button to create a new task exam and display the new task exam created. - Able to set a reminder and receive reminders for each task exam on time. - Able to click the title of the task exam to update task exam information and it will display changes made. - Able to delete any task exam by clicking the "Delete" icon and it will be removed from the list of task exams.	- When clicking the "Exam" button, it redirects students to the wrong page. - When clicking the "plus" button, students can not input the task exam information, or/and the new task exam created is not displayed. - Unable to set the reminder for each task exam or/and receive reminders before or later than the time and date set or do not receive reminders at all. - Does not allow the student to update task exam information or/and updates made did not display. - Failure to delete the task exam or/and the task exam deleted did not remove from the list of task exam.

REQ-006	Manage task: Event	Set task: Event reminder	- Students are redirected to the task event page and the list of task events is displayed when the student clicks the “Event” button. - Able to create a new task event by clicking the “plus” button, and the new task event created will be displayed. - Able to set a reminder and receive reminders for each task event on time. - Able to click the title of the task event to update task event information and it will display changes made. - Able to click the “Delete” icon to delete a task event and it will be removed from the list of task event.	- When clicking the “Event” button, it redirects students to the wrong page. - Students can not input the task event information when clicking the “plus” button, or/and the new task event created is not displayed. - Unable to set the reminder for each task event or/and receive reminders before or later than the time and date set or do not receive reminders at all. - Does not allow the student to update task event information or/and updates made did not display. - Unable to delete the task event or/and the deleted task event is not removed from the list of task events.
REQ-007	Manage task: Exercise	Set task: Exercise reminder	- A list of task exercises is displayed when the student clicks the “Exercise” button. - Able to create a new task exercise by clicking the “plus” button, and the new task exercise created will be displayed.	- Redirected student to the wrong page when clicking the “Exercise” button. - When clicking the “plus” button, students can not input the task exercise information, or/and the new task exercise created is not displayed.

			- Able to set a reminder and receive reminders for each task exercise on time. - Able to click the title of the task exercise to update task exercise information and it will display changes made. - Able to delete any task exercise by clicking the "Delete" icon and it will be removed from the list of task exercise.	- Unable to set the reminder for each task exercise or/and receive reminders before or later than the time and date set or do not receive reminders at all. - Does not allow the student to update task exercise information or/and not display the changes made. - Unable to delete the task exercise or/and the task exercise deleted did not remove from the list of task exersice.
Subsystem 2: My Social Network (MyCircle)				
REQ-008	Set basic information and social interest.	-	- Able to insert and save their basic information and social interest into the database. -Successfully display the student's basic information on their social profile.	- Failure to insert and save their basic information and social interest into the database. -Failure to successfully display the student basic information at their social profile.
REQ-009	View and Update Social Profile.	-	- Able to view their social profile which includes information about their participation in social clubs, marks earned from social activities and overall level of social engangment. - Able to update	- Failure to view their social profile which includes information about their participation in social clubs, marks earned from social activities and overall level of social engangment. - Failure to update their basic

			their basic information and social interests.	information and social interests.
REQ-0010	Search Friend	- Display Searched Friends - Add Friends	- Able to view the students using this system. - Able to search for the specific student they want to be friends with. - Able to use the "Add" feature to add the specified friend to be their friends and include them to their social circle.	- Failure to view the students using this system. - Failure to search the specific student they want to be friends with. - Failure to use the "Add" feature to add the specified friend to be their friends and include them to their social circle.
REQ-011	View Social Recommendations	Join Social Clubs	- The social recommendation is able to identify suitable social clubs for the students based on their interest. - The students are able to use the join feature to become a member of the social club they are interested in.	- The social recommendation is unable to identify suitable social clubs for the students based on their interest. - The students are unable to use the join feature to become a member of the social club they are interested in.
REQ-012	View Social Events.	- Participate in Social Events - Obtain Participation Marks	- Correct displays of social events based on social clubs are displayed. - Students are redirected to the correct social event they have chosen. - Students are able to fill up the participation form and are successfully	- Incorrect display of social events based on social clubs are displayed. - Students are redirected to the wrong social event they have chosen. - Students are not able to fill up the participation form and are not

			saved into the database. - Students are correctly awarded a certain amount of participation marks based on the level of involvement in the social event.	successfully saved into the database. - Students are incorrectly awarded a certain amount of participation marks based on the level of involvement in the social event.
REQ-013	Engage with Friends via Chat	-	- The system is able to open the correct chat window with the selected person - Able to send message to their friends. - Able to receive replies from their friend. - "Exercise with Friend" button works as expected where it successfully redirects the student to the page.	- The system is not able to open the correct chat window with the selected person - Unable to send messages to their friends. - Unable to receive replies from their friend. - "Exercise with Friend" button does not work as expected where it doesn't successfully redirect the student to the page.
Subsystem 3: Fitness Track and Healthy Eating (beFit)				
REQ-014	Manage Food Intake and Exercise Diary	-	- The student is able to add new food or exercise entry into the diary by clicking the "Add Food" or "Add Exercise" buttons respectively and display them under correct category; food or exercise - The student is able to save the food or exercise by clicking	- When clicking the "Add Food" or "Add Exercise" buttons, student cannot input any food or exercise or/and the entries are display under wrong category - After clicking the button "Save", it does not save the entries

			the “Save” button and the system successfully display them - The student is able to update the food or exercise by clicking the food or exercise entry in the diary respectively - The student is able to save changes made after editing the entries and the system successfully display them - The system is able to calculate the daily calorie intake goal after inputting food and exercise and display them	thus the system cannot successfully display the entries - After clicking the food or/and exercise entries in the diary, it does not direct the student to the edit page - After clicking the button “Save” after made changes, it does not save the entries thus the system cannot successfully display the entries - Failure to calculate or wrong calculation of the daily calorie intake goal after inputting food and exercise that results in displaying wrong post-calculation
REQ-015	Set Mealtimes Reminder	-	- The student is able to set time for all mealtimes - The system is able to alarm the student based on the time set punctually	- Failure to set time for all mealtimes or can only set time less than all mealtimes - The alarm rings later than the time set
REQ-016	View Real-Time Route with Calculated Steps	-	- The student is able to view the route on a map - The system is able to track and calculate steps, distance travelled and calories burned	- No route is displayed - No steps are tracked and calculated, and no distance and calories burned are calculated

REQ-017	Collect Points	Get Rewards	- The student is able to view list of goals set by the system after clicking the "Rewards" button in the Dashboard - The student is able to collect points after completing goals - The system is able to calculate the total points correctly - The student is able to get rewards after collecting sufficient points	- After clicking the "Rewards" button, it directs the student to other pages other than the list of goals or display blank page - Failure to get any points after completing goals by no achieved goals with its points is displayed - Wrong calculation of the total points like mistakenly deduct or add points - The system fails to reward the student even after collecting sufficient point
REQ-018	Create Exercise with Friends	-	- The student is able to add friends from the group chat by clicking the "Add" button - The student is able to click the "Save" button to save to the Task: Exercise - The system is able to save the exercise from the Task: Exercise to the Food Intake and Exercise Dairy under exercise and then display it	- After clcikng the "Add" button, no list of friends is displayed so failure to add friends - After clicking the "Save", the exercise is not displayed thus making it not saved to the Task: Exercise - Failure to retrieve any data from the Task: Exercise thus fail to save and display the entry

Table 4.3: List of item pass or fail criteria

REFERENCES

1. Discussion on SDG 4 – Quality education. Retrieved from https://sustainabledevelopment.un.org/content/documents/23669BN_SDG4.pdf

2. Scheiner, M. (2022, June 30). What is Agile Project Management Methodology? Basic Process Explained.CRM.org. Retrieved from https://crm.org/news/agile-project-management
3. Peak, S. (2022, Jun 29). What Is Agile Scrum Methodology? , *Business News Daily*. Retrieved from https://www.businessnewsdaily.com/4987-what-is-agile-scrum-methodology.html
4. Drummond, C. (n.d). Scrum A guide to scrum: what it is and how it works. Retrieved from https://www.atlassian.com/agile/scrum

5. C, Massimo. (2021, May 19). UML for Android Engineers. Retrieved from https://www.kodeco.com/21792733-uml-for-android-engineers

6. Class diagrams. (2021, 5 March). Retrieved from https://www.ibm.com/docs/en/rsm/7.5.0?topic=structure-class-diagrams

7. Activity Diagram. (n.d.). Retrieved from https://www.smartdraw.com/activity-diagram/

8. AOSP Java code style for contributors. (2023, 4 January). Retrieved from https://source.android.com/docs/setup/contribute/code-style
9. My Study Life- Online Student Planner. Retrieved from https://www.mystudylife.com/

10. Strickland, J. (n.d.). How google calendar works. HowStuffWorks. Retrieved from https://computer.howstuffworks.com/internet/basics/google-calendar.htm

11. Ajitsaria, A. (n.d). Build a Recommendation Engine with Collaborative Filtering Retrieved from https://realpython.com/build-recommendation-engine-collaborative-filtering/
12. Shekar, Y. (2022, May 13). Collaborative Filtering - ML Retrieved from https://www.geeksforgeeks.org/collaborative-filtering-ml/
13. Skovhøj, F (2022, Feb 9). Using Collaborative Filtering in E-Commerce: Advantages & Disadvantages. Retrieved from https://www.clerk.io/blog/collaborative-filtering
14. Daburra, Imad. (2018, Sept 18) . K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks. Retrieved from https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a
15. Bjarnadottir, A & Lang,A. (2022, September 28). The 8 Best Calorie Counter Apps of 2022. Retrieved from https://www.healthline.com/nutrition/5-best-calorie-counters
16. Hall, C. (2023, 1 January). What is Strava, how does it work and is it worth paying? Retrieved from for?https://www.pocket-lint.com/apps/news/154854-what-is-strava-and-how-does-it-work
17. How Distance is Calculated. (2022, 17 September). Retrieved from https://support.strava.com/hc/en-us/articles/216919487-How-Distance-is-Calculated
18. Project Requirement Gathering: Surveys and Questionnaires. (n.d.). Retrieved from https://www.pmmajik.com/project-requirement-gathering-surveys-and-questionnaires/

19. Pohl, K & Rupp, Chris. (2015). *Requirements Engineering Fundamentals*. Rocky Nook.
20. Domain Modeling Class Diagrams. (2021, March 16). Retrieved from https://www.ibm.com/docs/en/radfws/9.6.1?topic=diagrams-domain-modeling-class
21. Momale, A. (2022, March 23). Goals of Software Testing. Retrieved From https://www.geeksforgeeks.org/goals-of-software-testing/
22. Yasar, K. (n.d).What is Software Testing? Definition, Types and Importance. Retrieved From https://www.techtarget.com/whatis/definition/software-testing
23. Parthiban, P. (2021, April 14). " 7 Reasons Why Software Testing is Important". Retrieved From https://www.indiumsoftware.com/blog/why-software-testing/
24. 5 Phases of Testing in Software (Plus Testing Life Cycle) (2021, April 27). (https://www.indeed.com/career-advice/career-development/phases-of-testing
25. Acceptance Testing | Software Testing. (2022, 15 June). Retrieved from https://www.geeksforgeeks.org/acceptance-testing-software-testing/

APPENDICES

This is the survey link that is used for gathering user's requirements for the development of our system.
https://docs.google.com/forms/d/e/1FAIpQLScprF2DKlIrTyL9EJfwAcIhMImywkJwzvYFykXCNu6TfF8wUg/viewform?usp=sf_link