



 slington college
(इस्लिङ्टन कलेज)

Module Code & Module Title
CS5002NI SOFTWARE ENGINEERING

Assessment Weightage & Type
35% Individual Coursework

Year and Semester
2021-22 Spring

Student Name: Abhishek Basnet

London Met ID: 19033650

College ID: np01cp4s200067

Assignment Due Date: 9th April 2022

Assignment Submission Date: 9th April 2022

Title: T-14 Training Academy

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded

Table of Contents

1. INTRODUCTION.....	1
(Planning, requirements modelling and analysis)	2
2. GANTT CHART	2
3. USE CASE MODEL	3
3.1. Use Case Diagram	5
3.2. High level use case description	6
3.3. Expanded Use case Descriptions	9
4. COMMUNICATION DIAGRAM	12
4.1. Collaboration Diagram	12
4.2. Sequence Diagram	14
5. CLASS DIAGRAM.....	15
6. FURTHER DEVELOPMENT	17
7. PROTOTYPES.....	19
8. CONCLUSION	37
9. REFERENCES.....	38

List Of Tables

Table 1: Expanded Description of login use case	10
Table 2: Expanded description of Register membership use case.....	11

List Of Figures

Figure 1: Gantt Chart.....	2
Figure 2: Use Case Diagram.....	5
Figure 3: Collaboration Diagram	13
Figure 4: Sequence diagram	14
Figure 5: Class Diagram.....	16

1. INTRODUCTION

This coursework is assigned to the students in order to go through the scenario where T-14, Training Academy requires a system for automating their business. So, this assignment covers object oriented analysis and design for T-14. Various planning, modelling and further development has been carried out to meet the requirements of the organization. The Gantt chart is used to represent the time it takes to complete certain tasks using the RUP (Rational Unified Process) methodology. The use case is then created to capture all of the development requirements including the functions. Similarly, the collaboration diagram, sequence diagram and class diagrams are all created to define and clarify the roles of the objects that perform a particular flow of events of a use case. At the final stage, a software prototype was created to sketch the appearance of the T-14 software system.

The T-14 Training academy, specializes in providing a football training platform for different age groups. Due to the pandemic, this academy was looking for carrying on their program through online by developing an automated system so that they can still run the training academy without lagging behind the business.

(Planning, requirements modelling and analysis)

2. GANTT CHART

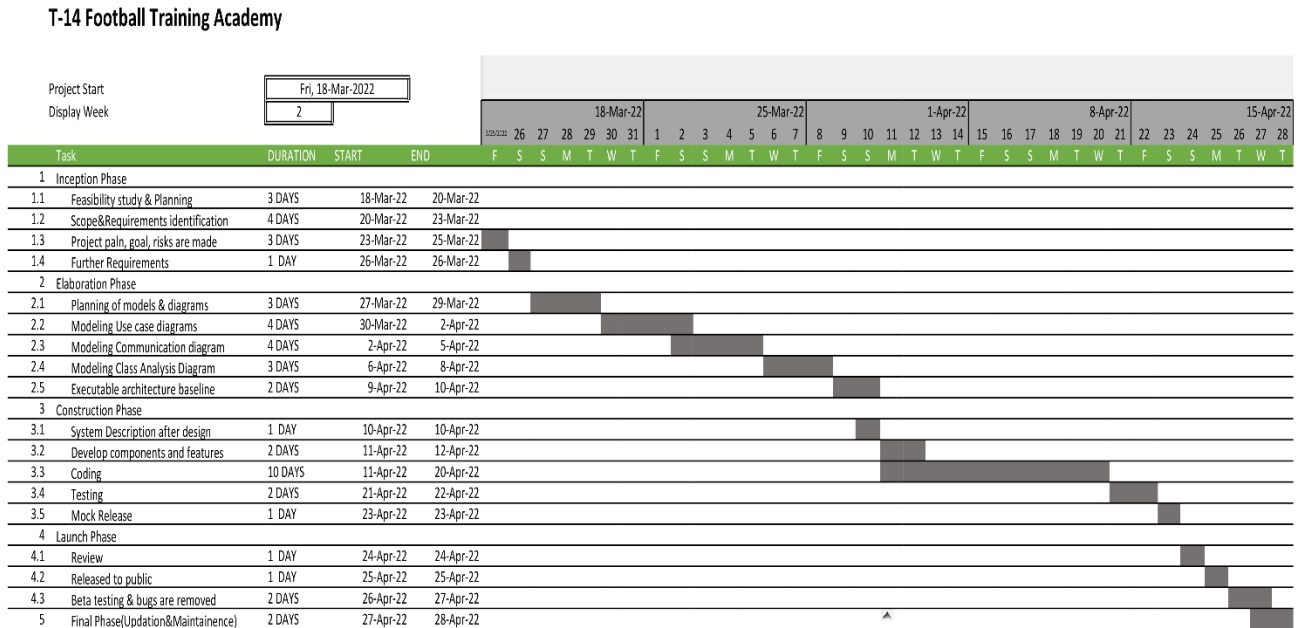


Figure 1: Gantt Chart

A Gantt chart is a commonly used graphical depiction of a project schedule. It's a type of bar chart showing the start and finish dates of a project's elements such as resources, planning and dependencies. Henry Gantt (1861-1919), an American mechanical engineer, designed the Gantt chart. The Gantt chart is the most widely used chart in project management. These charts are useful in planning a project and defining the sequence of tasks that require completion. In most instances, the chart is displayed as a horizontal bar chart. Horizontal bars of different lengths represent the project timeline, which can include task sequences, duration, and the start and end dates for each task. (Grant, 2021)

3. USE CASE MODEL

A Use Case Model describes a new system's projected functionality. A Use Case is a discrete unit of interaction between a system and a user (human or computer). This interaction, such as Create Account or View Account Details, is a single unit of meaningful work. (Sparx Systems, n.d.)

Use case diagram is composed of following:

System:

Draw your system's boundaries using a rectangle that contains use cases. Place actors outside the system's boundaries.

**Use Case:**

Draw use cases using ovals. Label the ovals with verbs that represent the system's functions.

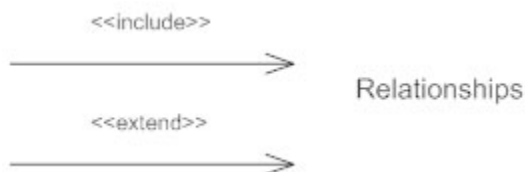


Actors

Actors are the users of a system. When one system is the actor of another system, label the actor system with the actor stereotype.

**Relationships**

Illustrate relationships between an actor and a use case with a simple line. For relationships among use cases, use arrows labeled either "uses" or "extends." A "uses" relationship indicates that one use case is needed by another in order to perform a task. An "extends" relationship indicates alternative options under a certain use case.



(smartdraw, 2022)

3.1. Use Case Diagram

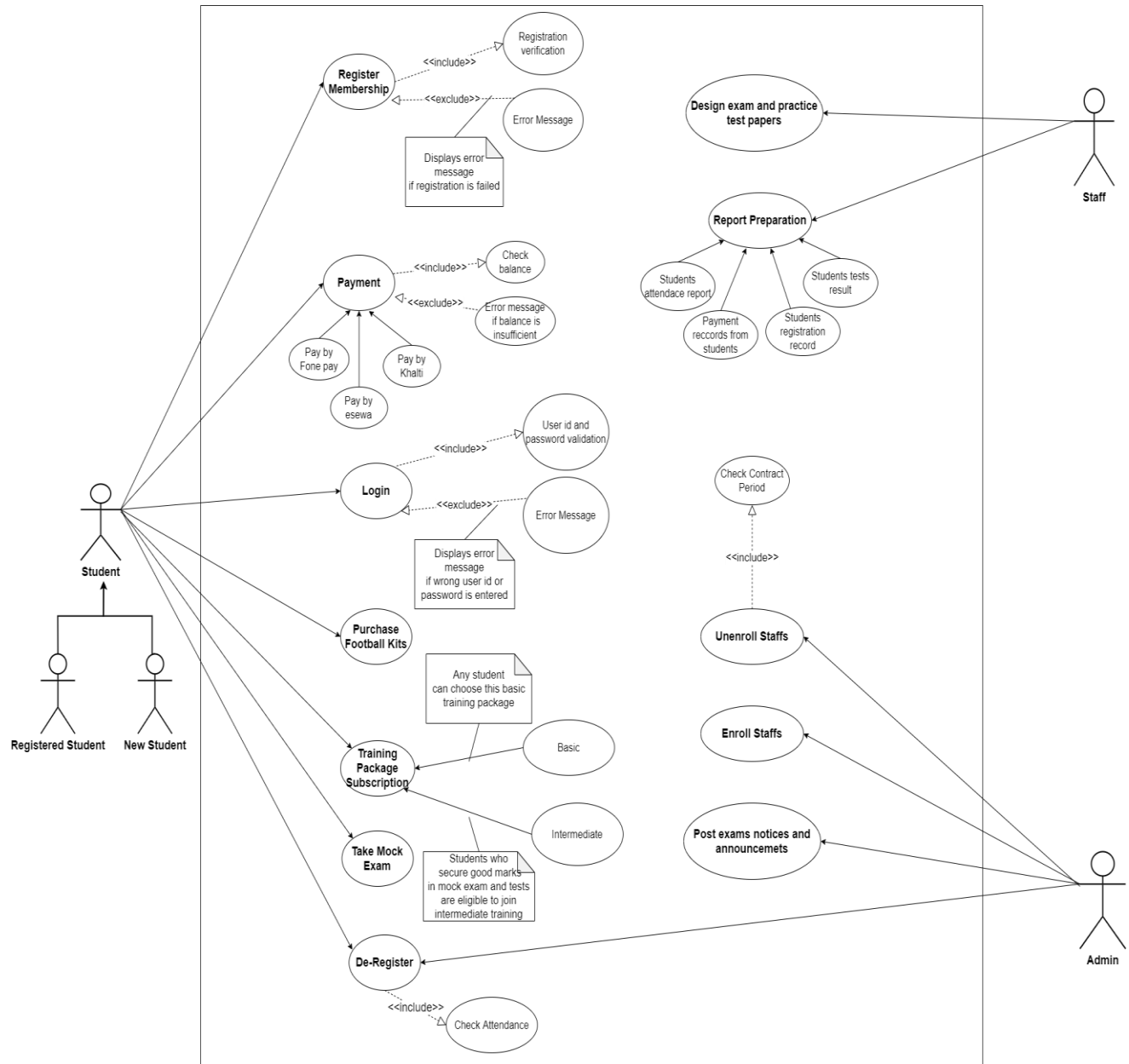


Figure 2: Use Case Diagram

3.2. High level use case description

a) Use case: Design exam and practice test papers

Actor: Staff

Description: The exam papers and practice test papers are designed by the staffs for the students who are the part of the training academy.

b) Use case: Report Preparation

Actor: Staff

Description: The report prepared by the staffs will include student's attendance report, payment records, student's registration records and tests result of the students.

c) Use case: Register Membership

Actor: Student

Description: The student's membership is registered after the individuals enter their valid details and accordingly pays for the registration.

d) Use case: Payment

Actor: Student

Description: The student makes payment and pays to the system to get registered in the system and then they are accessed to use the features available.

e) Use case: Login

Actor: Student

Description: The student logs in after successful completion of registration and is able to access the training course and give exams.

f) Use case: Purchase football kits

Actor: Student

Description: The students/members of the training academy, are able to purchase football kits in a discounted price which is a lot lower than market value from the system.

g) Use case: Training Package Subscription

Actor: Student

Description: Training package includes both basic and intermediate training. All the students can choose basic training package but the intermediate training package can be chosen only by the students who pass the mock exam.

h) Use case: Take Mock Exam

Actor: Student

Description: Students are able to take mock exam in order to join intermediate training package.

i) Use case: De-register

Actor: Student and Admin

Description: The student is able to de-register from the program. Whereas Admins can de-register the students whose attendance is very poor.

j) Use case: Enroll staffs

Actor: Admin

Description: The admin enrolls the staff to run the online platform and provide specific training to the students.

k) Use case: Un-enroll staffs

Actor: Admin

Description: The admin is able to unenroll the staff if their contract is finished or if the staff don't work according to contract.

I) Use case: Post exam notices and announcements

Actor: Admin

Description: The admin posts exam notices and announcements such as result and break notices in order to notify the students.

3.3. Expanded Use case Descriptions

a) Use case: Login

Actors: Student

Description: The student logs in after successful completion of registration and is able to access the training course and give exams.

Typical course of events:

Actor Events	System response
1. The successfully registered students proceeds to login in order to use the system for their training program	
	2. The system validates user ID and password entered by student, and then continues to give access to training package after mock exam if the user id and password are correct.

3. Student does tasks and various trainings according to the training package after getting logged in.	
	4. System stores the tasks done by the students and then provides feature for students to logout after the objectives are done.

Table 1: Expanded Description of login use case

Alternative courses:

Line 1:

Students cannot login if they are not registered yet.

Line 2;

The students may enter invalid details, and hence they will not be able to login.

b) Use case: Register membership

Actors: Student

Description: The student's membership is registered after the individuals enter their valid details and accordingly pays for the registration.

Typical course of events:

Actor Events	System response
1. Student enters personal details required for membership registration.	
	2. Verifies the information provided by the student and takes to payment section.
3. Student then fills the payment details and pays the required amount for registration.	
	4. Verifies payment and then registers the student.
5. Finally student gets registered and gets access to login.	

*Table 2: Expanded description of Register membership use case***Alternative courses:**

Line 1: If the student enters wrong username and password, then the student cannot continue registration for the membership as he/she will get error message.

Line 3: If the payment is not made, it will not accept the registration.

4. COMMUNICATION DIAGRAM

4.1. Collaboration Diagram

A Collaboration is a collection of named objects and actors with links connecting them. They collaborate in performing some task. Collaboration diagrams are used to show how objects interact to perform the behaviour of a use case or a part of a use case. Designers employ collaboration, to define and clarify the roles of the objects that perform a certain flow of actions in a use case. It is one of the most important source of information for determining class responsibilities and interfaces. (Visual Paradigm, 2022)

- **Objects:** The representation of an object is done by an object symbol with its name and class underlined, separated by a colon.
- **Actors:** In the collaboration diagram, the actor plays the main role as it invokes the interaction. Each actor has its respective role and name. In this, one actor initiates the use case.
- **Links:** The link is an instance of association, which associates the objects and actors. It portrays a relationship between the objects through which the messages are sent. It is represented by a solid line. The link helps an object to connect with or navigate to another object, such that the message flows are attached to links.
- **Messages:** It is a communication between objects which carries information and includes a sequence number, so that the activity may take place. It is represented by a labelled arrow, which is placed near a link. The messages are sent from the sender to the receiver, and the direction must be navigable in that particular direction. The receiver must understand the message.

(Java T Point)

In order to create a collaboration diagram, register membership (use case) has been used. The domain classes of the use case, Register membership are:

Student, payment and finally register.

Here,

- Actor: Student
- Boundary Object: : RegisterMembershipUI
- Controller Object: : RegisterMembership
- Other Objects: : Payment, : Student, : Register

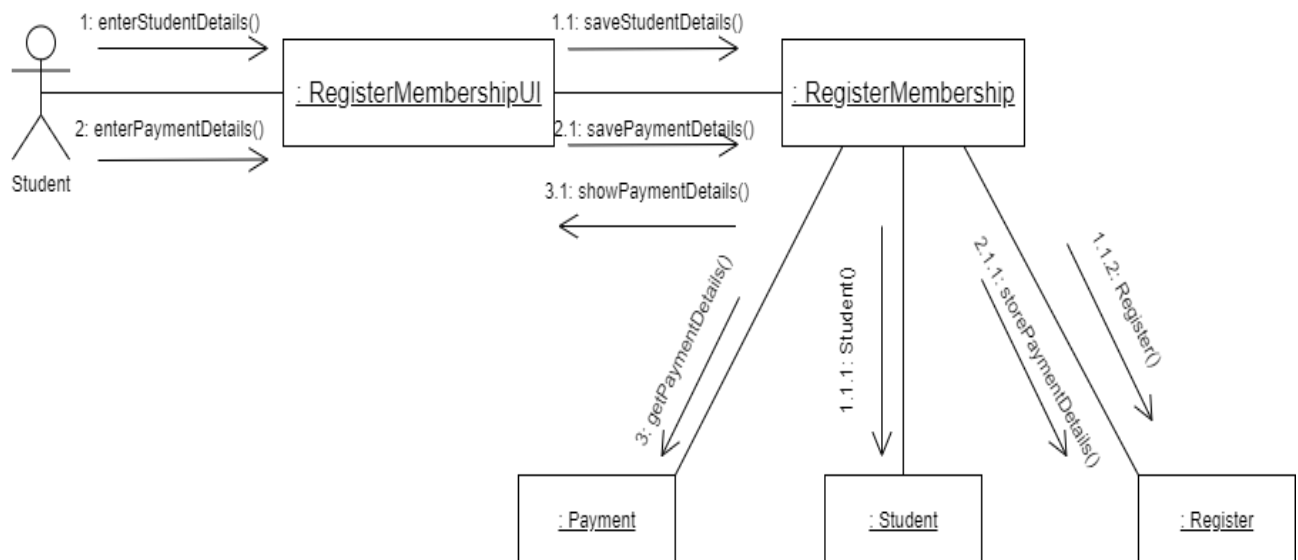


Figure 3: Collaboration Diagram

4.2. Sequence Diagram

The most typical interaction diagram is a sequence diagram. An interaction diagram is a diagram that depicts a system's interacting behaviour. We utilize several types of interaction diagrams to capture various features and components of interaction in a system since visualizing the interactions in a system can be difficult. Sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. A sequence diagram can also be referred to as an event diagram or an event scenario. Sequence diagrams show how and in what order the components of a system work together. Businessmen and software engineers often use these diagrams to document and understand requirements for new and current systems. (GeeksforGeeks, 2022)

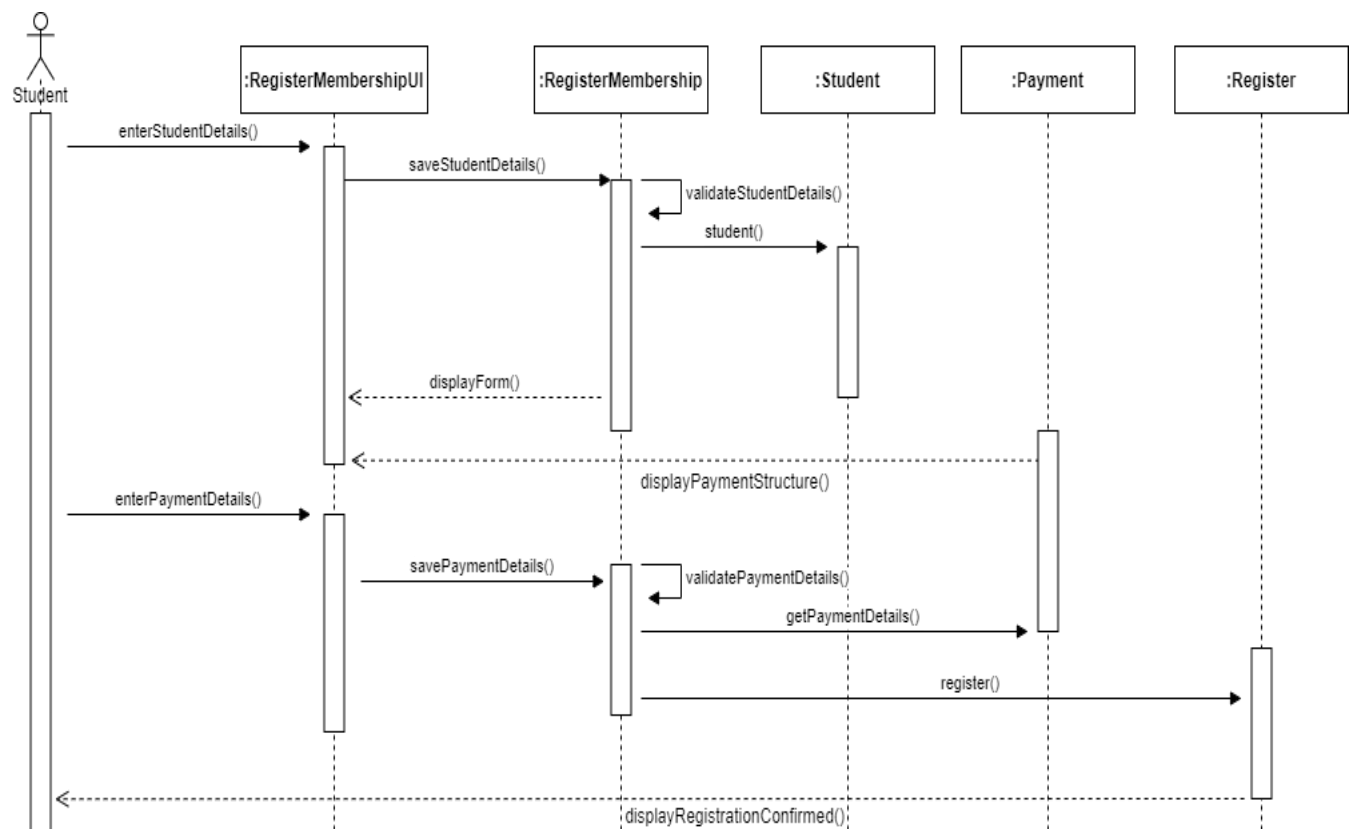


Figure 4: Sequence diagram

5. CLASS DIAGRAM

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. They show a collection of classes, interfaces, associations, collaborations, and constraints which can be known as a structural diagram. (Tutorialspoint, n.d.)

Class diagram of the system is shown below:

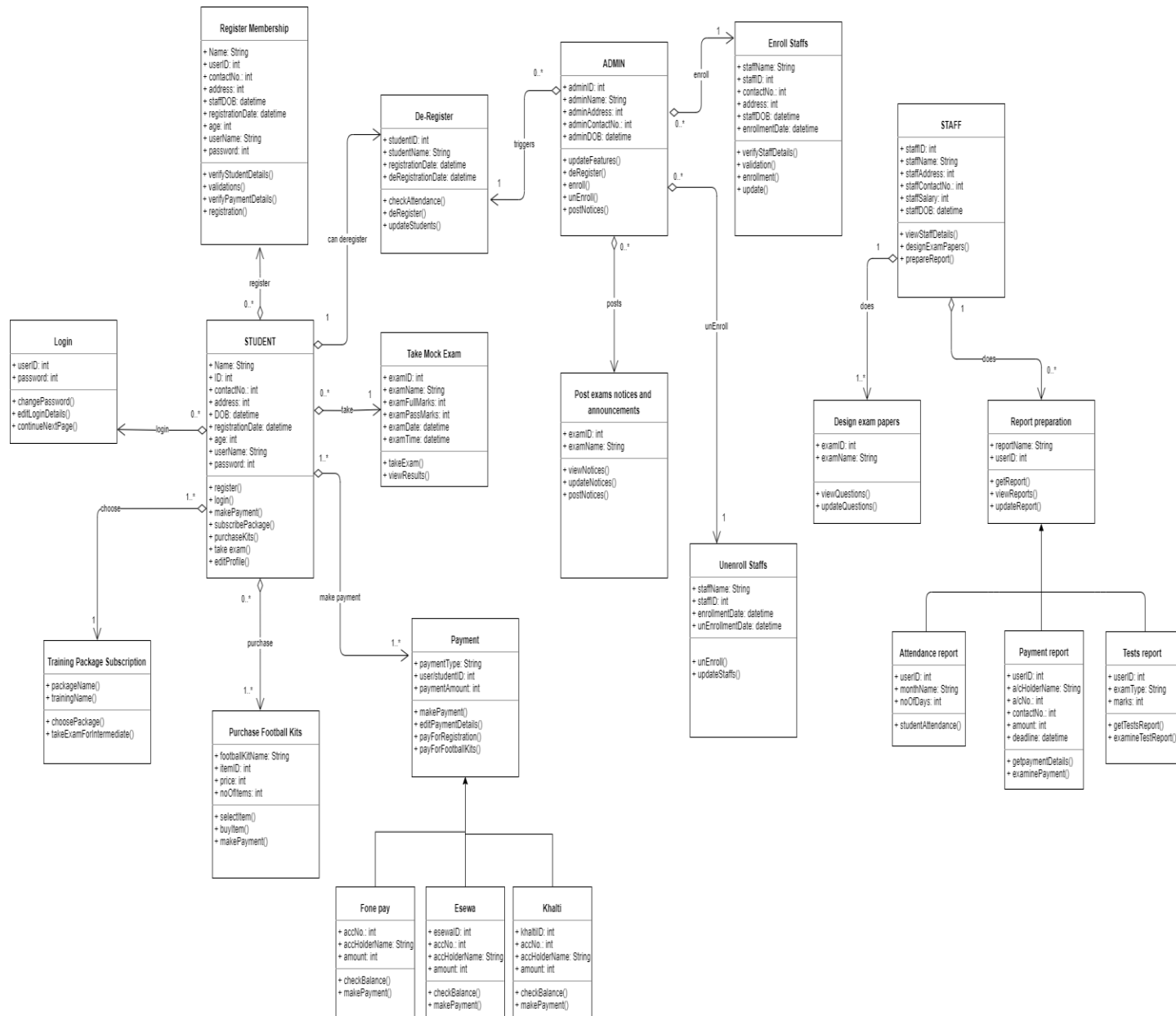


Figure 5: Class Diagram

6. FURTHER DEVELOPMENT

After the completion elaboration process which encompasses all of the design and planning work. The further development of the system are mentioned below:

- **Coding:**

After pointing out the required functions and features coding is done and implemented for the system to work. Developers and code specialists can use their coding in order to make the system run with various features. As mentioned in Gantt chart, coding can take up to 10 days for the developers. To create a user-friendly system, simple and understanding codes are required which will be easy to update. Developer can use any software to code for the system. Some of the software used for coding can be: Java, Netbeans, Python, etc. Codes which are not useful for the system shall be eliminated in order to manage the system. Comments for the codes and commands can be helpful to manage, edit, update, codes later. Errors should be handled well in order to run the system effortlessly.

- **Testing:**

The completion of coding will then take us to the testing part. Testing is a process for verifying whether the actual software or system product matches the expected requirements and ensuring that it is defect-free. Testing is vital because it uncovers defects/bugs before the software is delivered to the client, ensuring the software's quality. It improves the software's reliability and ease of use. Software that has been thoroughly tested ensures that it runs reliably and efficiently. A properly tested system product ensures dependability, security, and high performance, which leads to time savings, cost effectiveness, and customer satisfaction. Requirement based testing,

compatibility testing, top-down integration testing, bottom-up integration testing, sandwich testing, regression testing, smoke testing etc. are some of the testing implemented after the coding. Similarly the three software testing are: Black box testing, Grey box testing and white box testing. These all testing will be done throughout various stages of development.

- **Instruction manual development:**

Instruction manual is a technical document intended to provide information and instruction on using a particular system. An instruction manual is prepared in order to guide the users regarding the system. The information is required for the users in order to run the system successfully. All the requirements to use the system is provided to the users. A good comprehensive manual consumes less time in explaining about the feature.

- **Normalization:**

Normalization is the process of reorganizing data in a database so that there is no redundancy of data, where all data is stored in only one place.

Data dependencies are logical and all related data items are stored together.

There are four steps of normalization and they are:

UNF (Un normalized Form): All repeating and non-repeating groupings are identified in UNF.

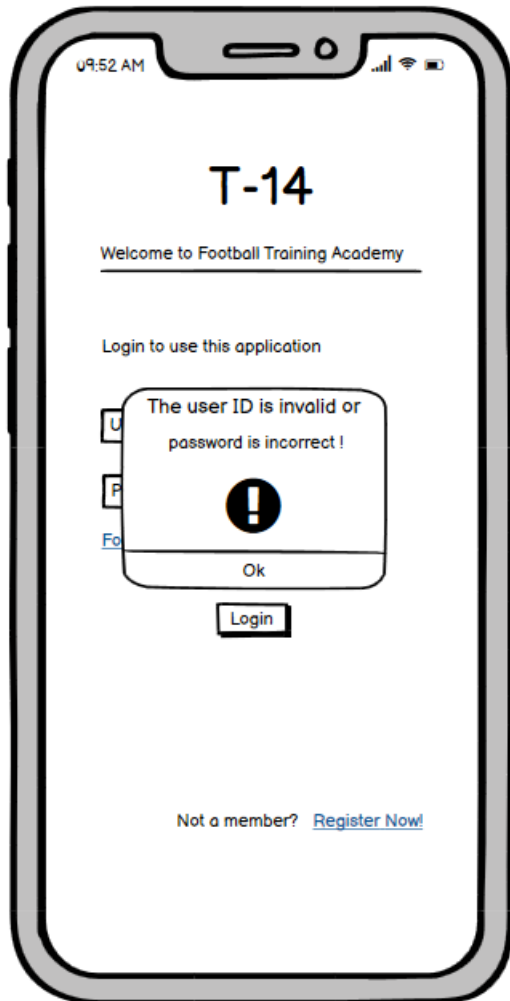
1NF: The repeating groups are transferred to a new entity in 1NF.

2NF: The functional dependency is verified, and the partial key is deleted in 2NF.

3NF: In 3NF, all non-key attribute dependencies within each table are discovered and transferred to a new table. A transitive dependency been found and transferred to a new table.

7. PROTOTYPES





The image shows a mobile application interface for a registration form. The screen is titled "Registration" and features a hamburger menu icon in the top left corner. The status bar at the top displays the time as 09:52 AM, along with signal, Wi-Fi, and battery icons. The form consists of the following fields:

- First Name
- Last Name
- Contact No.
- Address
- Email ID
- Date Of Birth (with a calendar icon)
- User ID
- Password (with an eye icon for toggling visibility)

Below the form fields is a "Register" button. At the bottom center of the screen is a circular back arrow button.

The image shows a mobile application interface for user registration. At the top, the status bar displays the time as 09:52 AM, along with signal and battery icons. The app's header includes a hamburger menu icon on the left and the title "Registration" in the center. Below the header, there are several input fields: "Abhishek" in the first field, "Basnet" in the second, and partially visible fields for "Con", "Add", "Em", and "Dat". A modal dialog box is overlaid on the form, displaying the message "Invalid details ! Please enter valid details without missing any fields" with a large exclamation mark icon. The dialog has an "ok" button at the bottom. Below the dialog, the "Avi123" field is visible, followed by a password field containing eight asterisks and a toggle icon. A "Register" button is positioned below the password field. At the very bottom of the screen is a circular back arrow icon.

09:52 AM

☰

Registration

Choose Payment Method


☐ Khalti

☒ Fonepay

☐ Esewa


A/C No.:

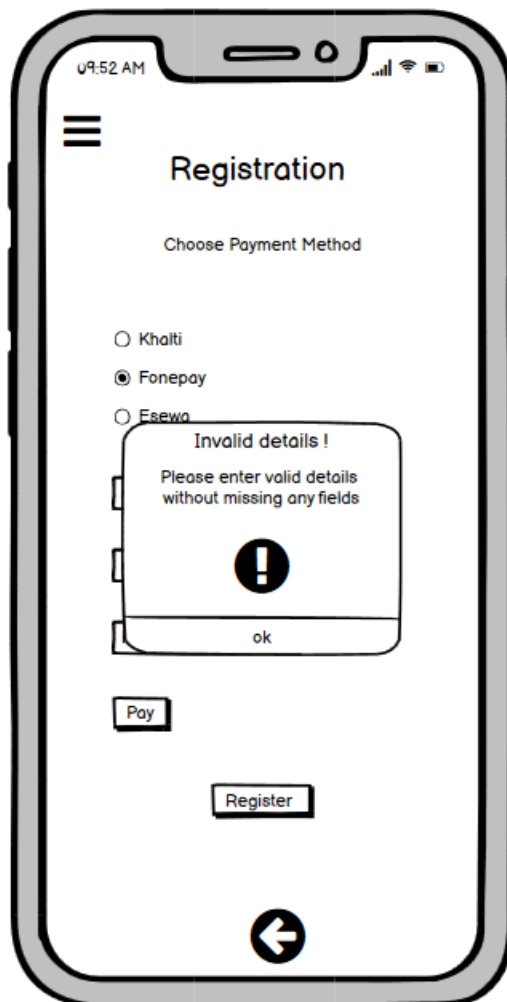
Amount

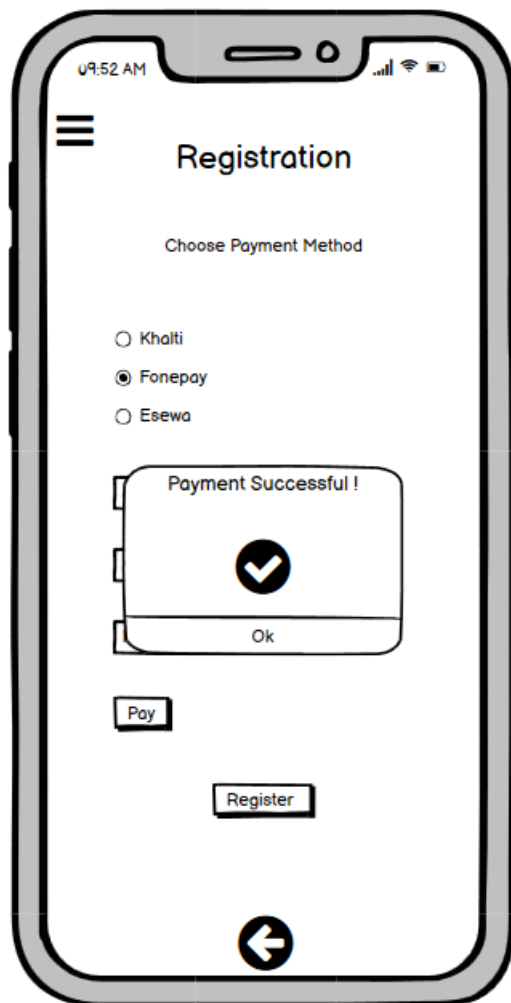
Pin Number 

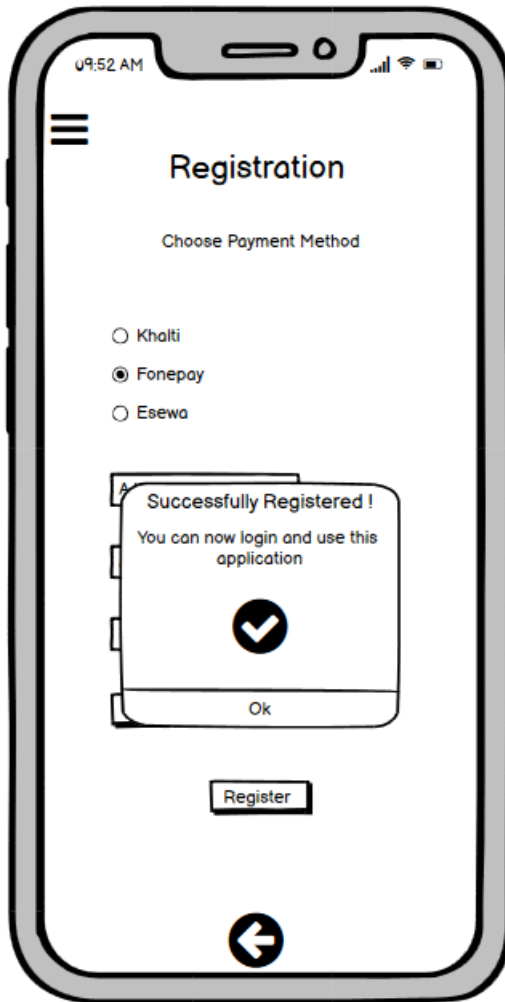
Pay

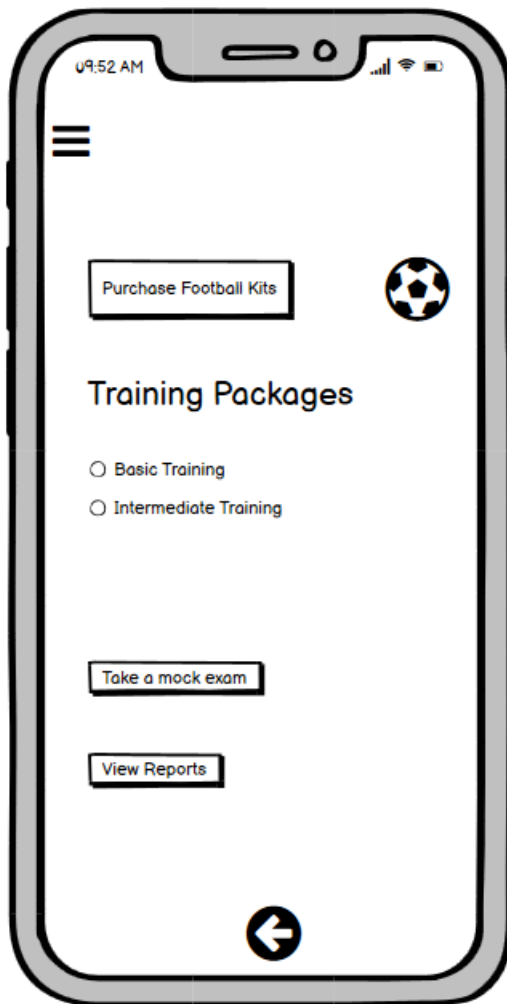
Register

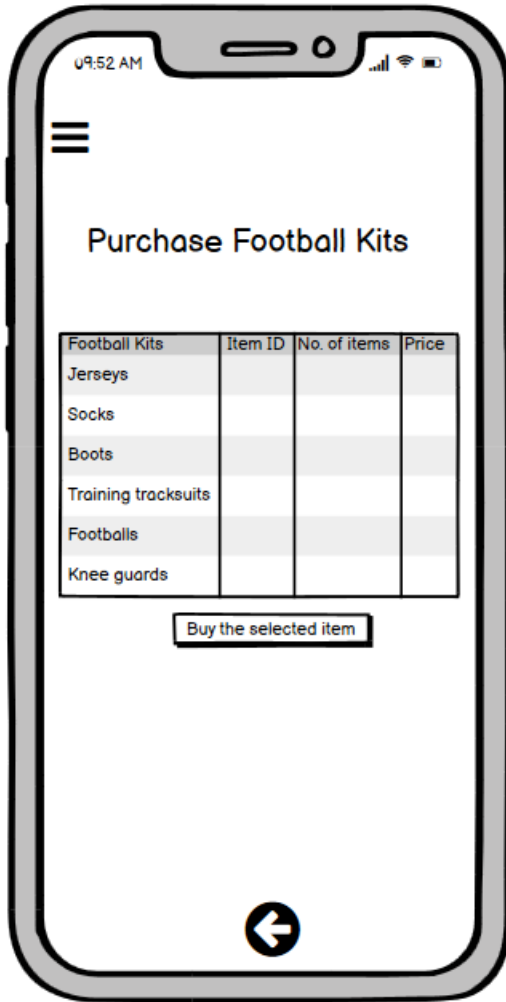


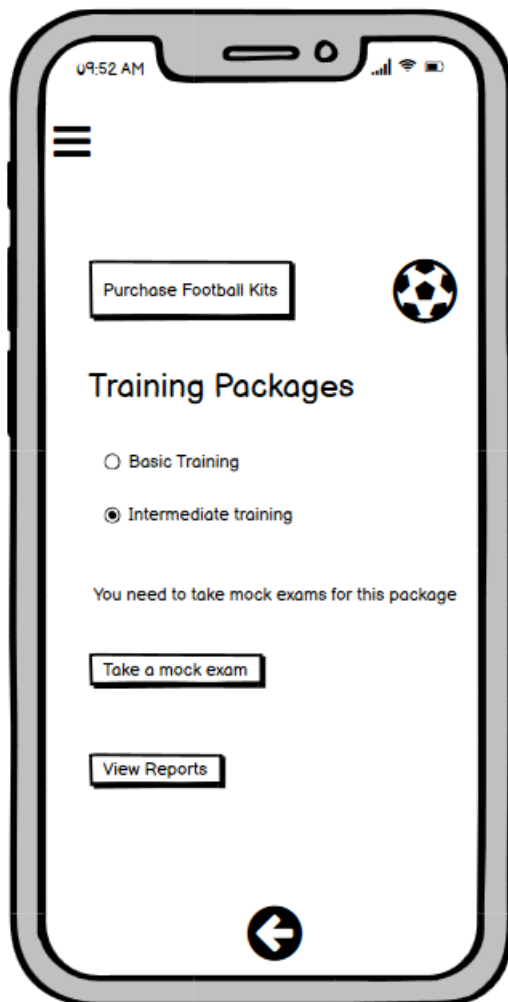






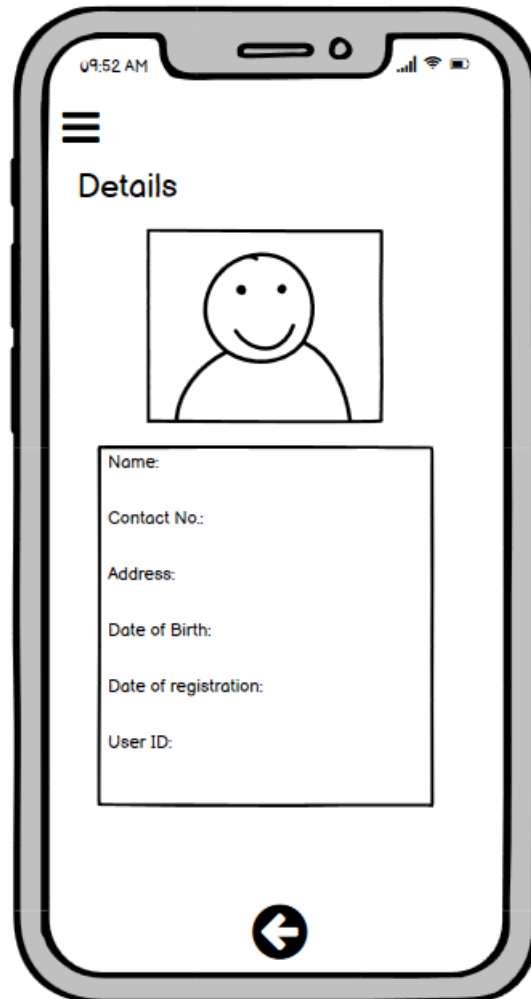
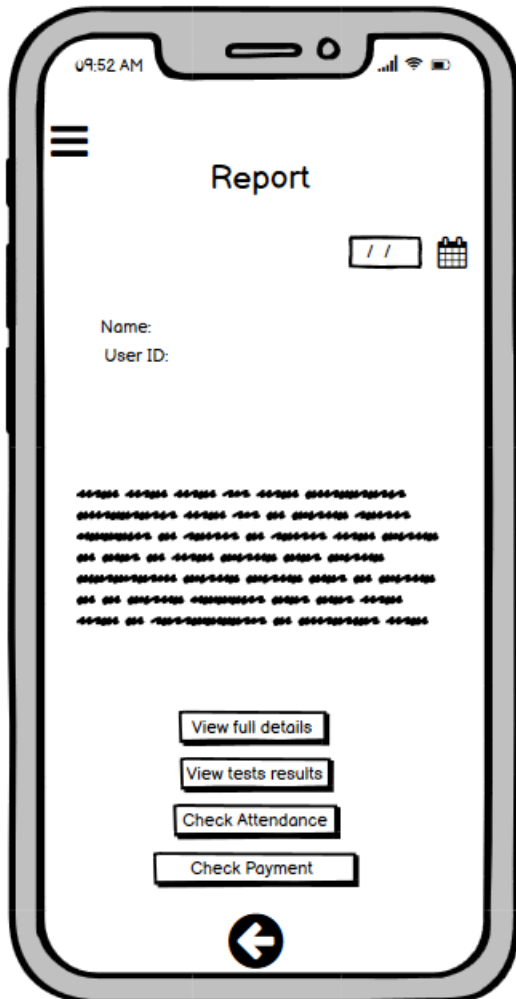


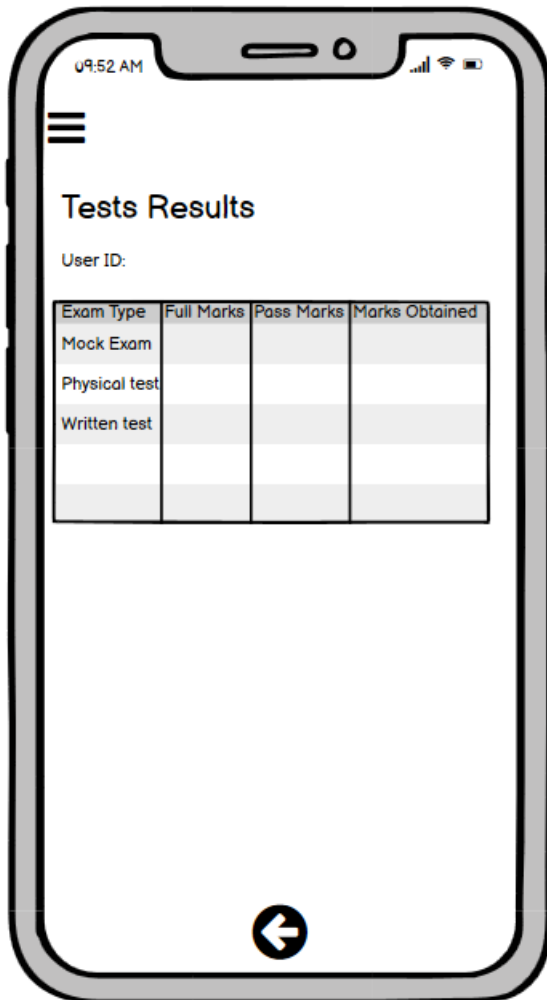












09:52 AM


☰

Tests Results

User ID:

Exam Type	Full Marks	Pass Marks	Marks Obtained
Mock Exam			
Physical test			
Written test			

←



09:52 AM

☰

ATTENDANCE

User ID:

1. January

Days Present:

Days Absent:

2. February

Days Present:

Days Absent:

3. March

Days Present:

Days Absent:

←

09:52 AM

☰

Payment Details

User ID:

A/C holder name:

A/C No.:

Contact No.:

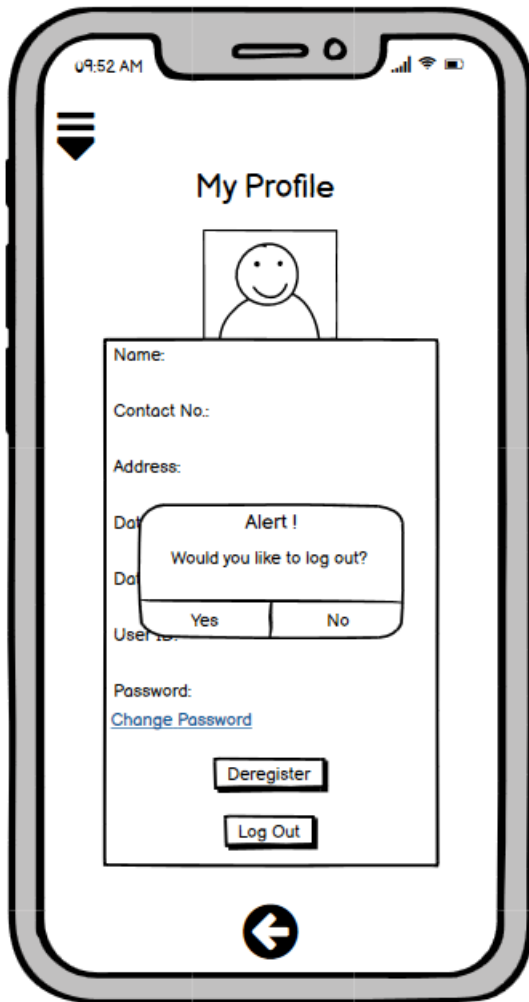
Due Amount:

Due Amount deadline:

Items	Payment type	Paid amount
Registration form		
Football Kits		

←





8. CONCLUSION

Just like the first coursework that had been assigned to us before, this second coursework is based on same scenario. This coursework mainly focuses on object oriented analysis and development. So, a proper functional system has been designed and developed for T-14 Football Training Academy. This report includes all the explanation of the development required for the system. At first, it was a great challenge to complete all the diagrams like Gantt Chart, Use case diagram, Communication diagram (Collaboration diagram and Sequence Diagram) ,as they were completely new for us. I was unsure about the completion of this coursework in time within the deadline. However I dealt with all the challenges by using my skills and efforts. I researched through various resources and asked my tutors which helped me a lot in handling the coursework.

We had a great opportunity to do this coursework because it helped us broaden up our mind with skills and knowledge regarding the management of any projects. I will definitely bring those knowledge in use in the future as well.

9. REFERENCES

- GeeksforGeeks*. (2022, May 6). Retrieved from <https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/>
- Grant, M. (2021, August 26). *Investopedia*. Retrieved from Investopedia: <https://www.investopedia.com/terms/g/gantt-chart.asp>
- Java T Point*. (2022). Retrieved from <https://www.javatpoint.com/uml-collaboration-diagram>
- smartdraw*. (2022). Retrieved from <https://www.smartdraw.com/use-case-diagram/>
- Sparx Systems*. (n.d.).
- Tutorialspoint*. (n.d.). Retrieved from https://www.tutorialspoint.com/uml/uml_class_diagram.htm
- Visual Pardigm*. (2022, 5 1). Retrieved from Visual Paradigm: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml-collaboration-diagram/>