



Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS5002N1 SOFTWARE ENGINEERING

**Assessment Weightage &
Type 35% Individual
Coursework**

Year and Semester

2019-20 Autumn / 2020-21 Spring

Student Name: Subarna Karki

London Met ID: 20049059

College ID: np01cp4s210089

Assignment Due Date: 9th May 2022

Assignment Submission Date: 9th May 2022

Title (Where Required): Coursework

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
2. Gantt chart.....	2
3. Use case Model.....	3
3.1. Use case diagram	4
3.2. High level Use case Description:.....	5
3.3. Expanded Use Case Description	8
4. Communication Diagram	10
4.2. Collaboration diagram	10
4.2. Sequence Diagram.....	12
4.3. Class diagram	14
5. Further Development	16
7. Prototype	18
8. Conclusion.....	31
Bibliography	32

Table of Figures

Figure 1: Gantt chart.....	2
Figure 2: Use case diagram.....	4
Figure 3: Collaboration Diagram	11
Figure 4: Sequence Diagram	13
Figure 5: Class diagram.....	15
Figure 6: Prototype of Login Form	18
Figure 7: Prototype of Login Error	19
Figure 8: Prototype of Registration Form	20
Figure 9: Prototype of Registration Error.....	21
Figure 10: Prototype of Payment Form	22
Figure 11: Prototype of Payment Error.....	23
Figure 12: Prototype of Payment success.....	24
Figure 13: Prototype of Mock Test	25
Figure 14: Prototype of Error in Mock test.....	26
Figure 15: Prototype of Closing Mock Test	27
Figure 16: : Prototype of mock Test Submission	28
Figure 17: : Prototype of Football kit showcase.....	29
Figure 18: Prototype of report generation	30

Table of Tables

Table 1: First Expanded use case description.....	8
Table 2: Second Expanded Use case description	9

1. Introduction

The coursework is grounded on a premise in which we must design an online platform for T-14 Football Academy that tracks all business as well as clients' payment records and progress in a well-structured and systematic manner. The object oriented analysis and design for their online system is addressed in this course. The use case model is used to explain how different categories of users interact with the system to solve an issue. Following the RAD methodology, a Gantt chart is utilized to plan the project's length and schedule. The sequence diagram is utilized to portray the transmission of communication from one object to the other, while the collaboration diagram is implemented to reflect the system's internal structure and the messages that are sent and received. Finally, a software simulation is formed to visualize the reality of the T14 football academy's online system.

T14 Football specializes in providing individuals with football training. The T-14 has been hindered by the Covid-19 epidemic since individuals were not permitted to meet physically, resulting in a 51% decline in enrolment, with just 457 persons enrolling this year compared to 800-900 in prior years, and present students are unable to take respective exams. The main goal of this program is to enhance the enrollment numbers population, that will significantly improve the company's revenue, by elongating their business on the internet, allowing users access to Academic's system, choose trainings based on user's age and preferences, and, most relevantly, making training safe for people since they can train from the comfort of their own homes.

2. Gantt chart

A Gantt chart is a graphical representation of a project schedule that is widely utilized. It's a form of bar chart that displays the start and end dates of project aspects including resources, plan, and relationships (investopedia, 2017).

Gantt charts are used to show the progression of several projects and programs that run concurrently inside a company. They are being used by managers to plan and schedule such projects in order for resources to be deployed optimally and for targeted initiatives to be completed prior less important ones commence.

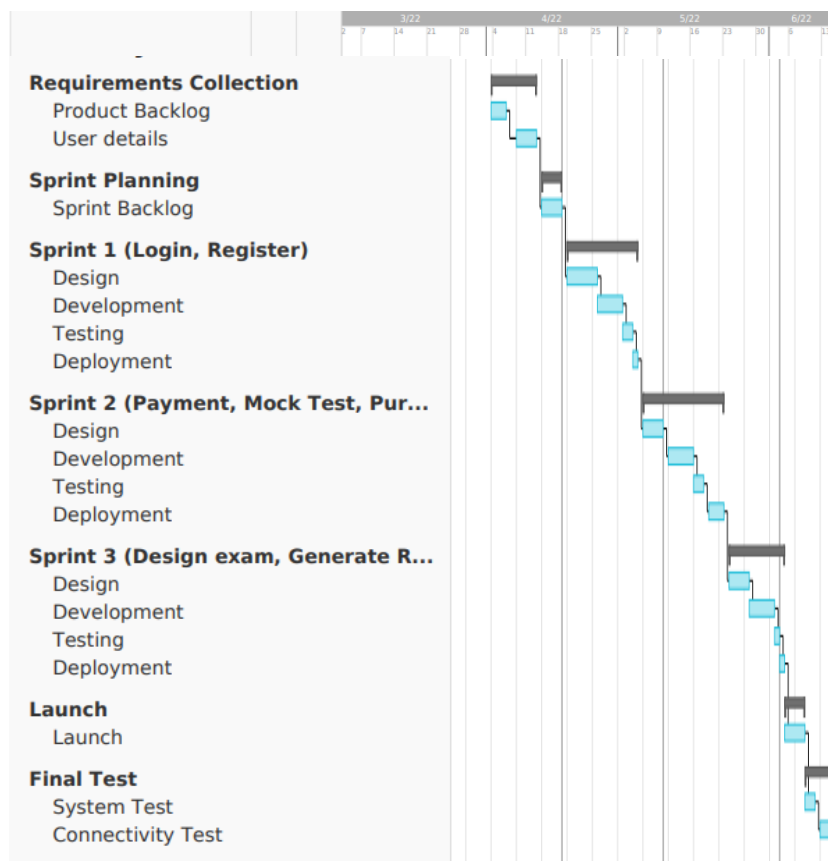


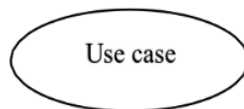
Figure 1: Gantt chart

3. Use case Model

The use-case model is a diagram that depicts how users interact with a system to resolve an issue. As a result, the use case model specifies the user's goal, the system's interaction with the user, and the system's actions necessary to achieve these goals (javatpoint, 2017).

Use case diagram symbols and notation:-

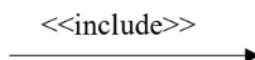
- Use case: Ovals with a horizontal shape that symbolize the various uses that a user could have.



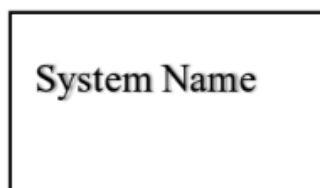
- Actors: People who are actually using the use cases are represented by stick figures.



- Relation: A relation is drawn between actors and use cases. It's crucial to know which actors are linked to which use cases in complicated graphs.



- System boundary box: A box that limits the scope of a system to use cases. Outside of the box use scenarios would be regarded outside the scope of the system.



3.1. Use case diagram

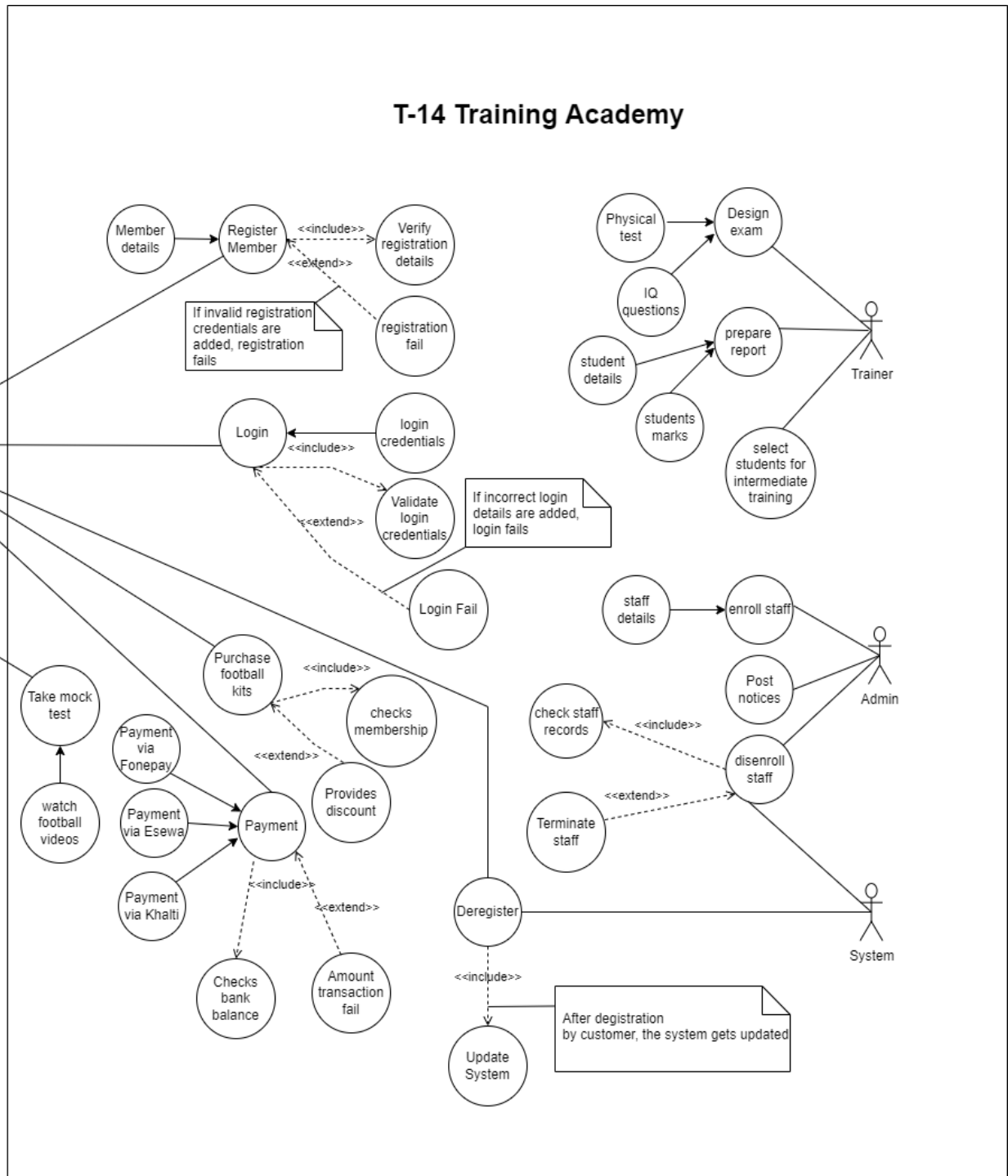


Figure 2: Use case diagram

3.2. High level Use case Description:

The high-level use case is essentially a brief explanation of the task stated in unstructured prose of one or two paragraphs. Its goal is to offer us only enough information to get a sense of the problem's complexity and to aid us in grouping similar use cases for creation in the Elaboration phase.

Use Case: Register Student

Actor: Student, Admin

Description: The new student gets registered in the Academy's online system by providing all of the needed registration information.

Use Case: Payment

Actor: Student, System (payment record)

Description: Students are only registered when payment is received, thus payment must be done first. The students can choose various means of payment like cash, cheque and online method of payment via FonePay, Khalti and Esewa.

Use Case: Login

Actor: Student

Description: By inputting all of the relevant credentials that were supplied to him/her during the registration process, the enrolled student gains access to the Academy's online system.

Use Case: Deregister student

Actor: Student, System

Description: For any justifiable cause, a registered user can deregister from the Academy's online system at any time.

Use Case: Take mock test

Actor: Student

Description: Students can take a mock tests before the exam that they have to pass to get enrollment in the Intermediate football Training.

Use Case: Purchase football kits

Actor: Student, System(checks the membership)

Description: The students can purchase all the required football kits like Jerseys, Socks, Boots, training tracksuits etc. within the academy and the registered students even gets the special discount.

Use Case: Design Exam Papers

Actor: Trainer

Description: The trainer designs the training papers for the students who have to take part in examination for enrolling themselves in intermediate training.

Use Case: Prepare Report

Actor: Trainer

Description: The trainer prepares the reports of all the students who have given the examination for enrolling themselves in intermediate training which include their marks and progress.

Use Case: Select students for Intermediate Training

Actor: Trainer, System

Description: The trainer is in charge of choosing the student for enrollment in intermediate training. The student's selection is based on the marks and examination results in his or her examination.

Use Case: Enroll Staff

Actor: Admin

Description: The Admin grants permission for staff to enroll in the Academy. The admin validates all of the details of the staff, including personal information and job experience, before allowing them to enroll.

Use Case: Disenroll staff

Actor: Admin

Description: The Academy's staff is disenrolled when the contract expires unless and until the contract is extended. If something goes beyond the contract, the Admin has complete right to dismiss the staff from the academy.

Use Case: Post Announcements

Actor: Admin

Description: The Admin has the responsibility to post all the notices and announcements in the Academy's online system so that users can remain update about the academy.

3.3. Expanded Use Case Description

Use case: Register student

Actor: Student, Admin

Description: The new student gets registered in the Academy's online system by providing all of the needed registration information.

Typical Couse of Events:

Student Action	System Response
1. A new student comes into the Academy and fills out a registration form.	2. The system verifies the registration form, and informs the student about the fee structure of the Academy.
3. The student completes the payment details and transmits the payment.	4. The Registration Form saved and store in the database by the system.
	5. The Academy Membership Document is certified and submitted to the new student.
6. With the Academy Member Document, the student leaves the system.	

Table 1: First Expanded use case description

Alternative course:

Line 1: The new student provides the invalid information in the registration form. Use Case ends.

Line 3: In the Payment details, the new student submits faulty credentials. Use Case ends.

Use Case: Enroll Staff**Actor:** Admin

Description: The Admin grants permission for staff to enroll in the Academy. The admin validates all of the details of the staff, including personal information and job experience, before allowing them to enroll.

Typical Course of Events:

Staff Action	System Response
1. The candidate completes the enrollment form and submits all relevant information.	2. The system validates the enrollment form and contacts candidate for an interview.
3. The candidate gives the interview, answering all of the questions and detailing the previous job experience.	4. The admin takes note of the responses and the candidate's expertise.
	5. The admin offers the contract to the candidate for the Academy.
6. The candidate signs the contract and becomes the staff of the Academy.	

Table 2: Second Expanded Use case description

Alternative Courses:

Line 1: The candidate submits the false information about himself/herself. Use Case ends.

Line 4: The Admin was satisfied with candidate's responses. Use Case ends.

4. Communication Diagram

4.2. Collaboration diagram

In the Unified Modeling Language, a collaboration diagram, commonly referred as a communication diagram, depicts the relationships and interactions among software components (UML). These diagrams may be used to depict the dynamic behavior of a certain use case and clarify each object's purpose (techtarget, 2019). The architectural elements necessary to ensure that the objectives of an interaction are first identified before creating a collaboration diagram. The relationships in between these parts are then used to build a model.

Usage of Collaboration Diagram:-

- (i) Model collaborations between objects or roles that give use case and operational functionality.
- (ii) Design mechanisms as part of the system's design and architecture.
- (iii) Observe interactions to see just how messages are sent between objects and roles in the collaboration.
- (iv) Construct distinct situations inside use cases or operations involving the collaboration of various objects and activities.

Steps for creating collaboration diagram:-

- (i) Expanded use case descriptions are created for the specified use case.
- (ii) Determine the behavior for which the realization and implementation are specified.
- (iii) Discover the structural elements of collaborative functionality, such as class roles, objects, and subsystems.
- (iv) Lines of association, as well as the use of multiple alternative scenarios in objects, are used to depict the interaction between objects.

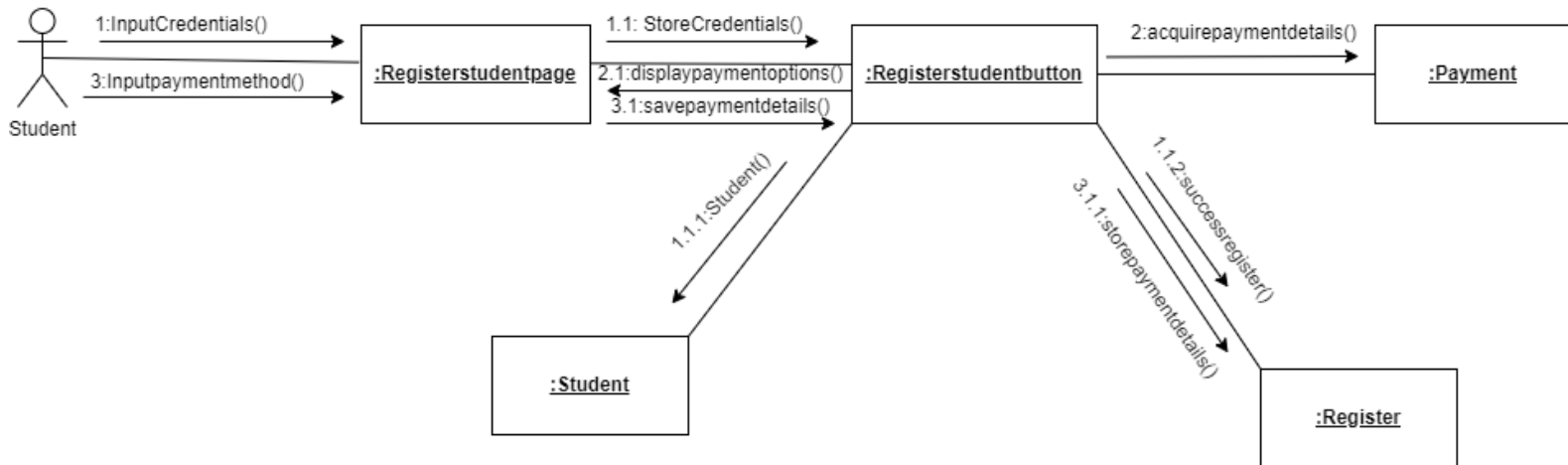


Figure 3: Collaboration Diagram

4.2. Sequence Diagram

A sequence diagram is a form of interaction diagram that shows how a set of objects interacts and in what order (lucidchart, 2018). Software engineers and business experts use these diagrams to comprehend the requirements for a new system or to describe an established system. The most common interaction diagram is a sequence diagram.

Usage of sequence diagram:-

- (i) Used to represent and visualize the reasoning underlying a complex function, method, or action.
- (ii) They're also used to demonstrate UML use case diagram information.
- (iii) Used to figure out how present or future systems work in detail.
- (iv) Visualize the flow of information and tasks among objects or components in a system.

Steps for creating sequence diagram:-

- (i) For the selected use case, expanded use case descriptions are generated.
- (ii) All of the suitable use cases are used to determine possible classes.
- (iii) All of the required classes are converted to objects.
- (iv) Activation bar, Timeline, and Lifeline are all configured on sequence diagram's objects.
- (v) To demonstrate the interaction, lines of association are constructed along with the utilization of numerous alternative scenarios in objects.

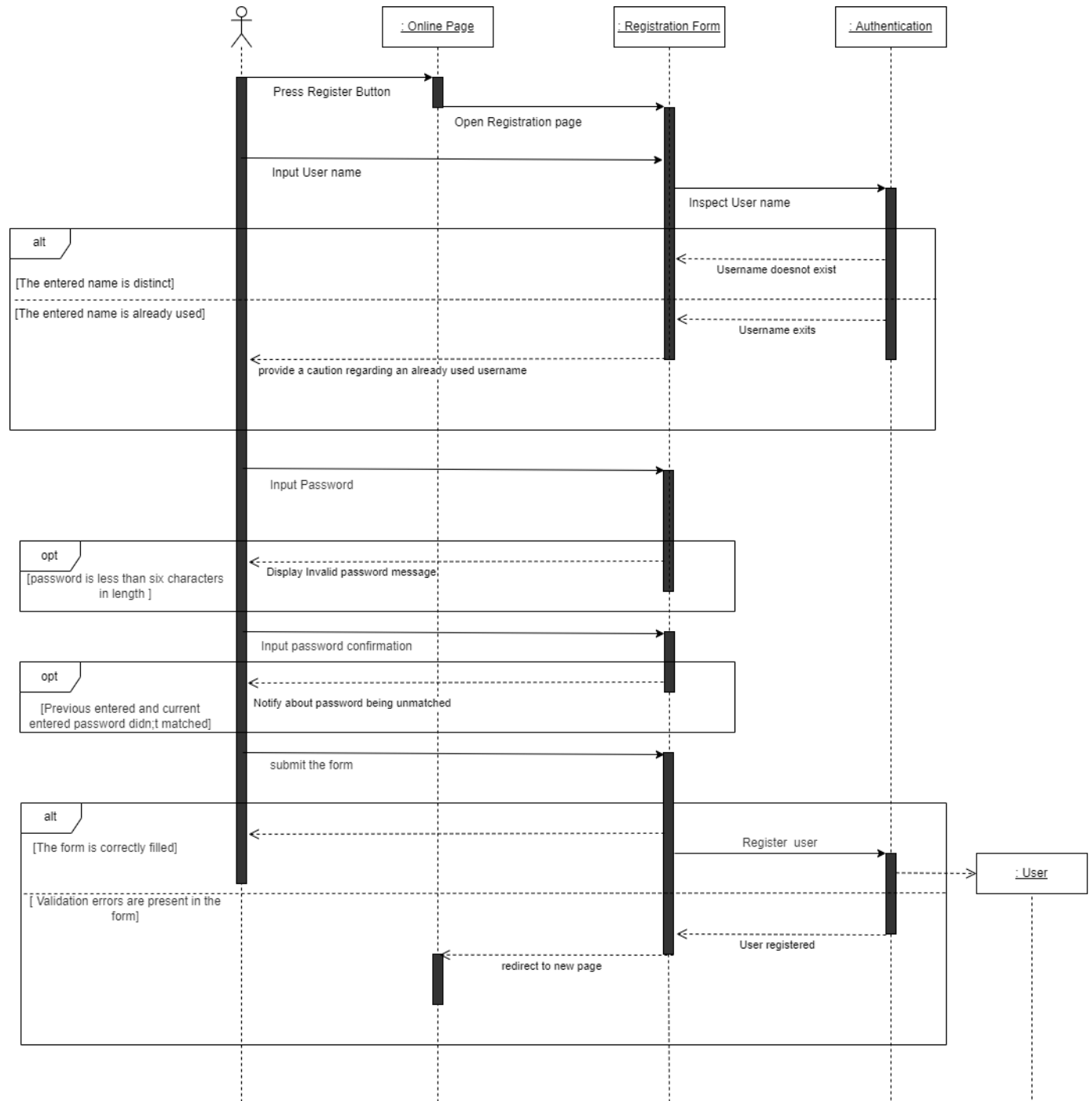


Figure 4: Sequence Diagram

4.3. Class diagram

A class diagram in the Unified Modeling Language (UML) is a form of static structural diagram in software engineering that depicts the system's structure by displaying the system's classes, properties, methods, and relationships between objects (visual-paradigm, 2018). It depicts a software's static view. A class diagram is used not just for viewing, characterizing, and describing many parts of a system, as well as for creating executable code for a software program.

Objectives of class diagram:-

- (i) Represents the system's static structure of classifications.
- (ii) Diagram serves as a foundation for additional UML-required structural diagrams.
- (iii) Also beneficial to programmers and other members of the team.
- (iv) Class diagrams can be used by industry experts to vitro models from a business standpoint.

Steps for creating a class diagram:-

- (i) All of the suitable use cases are used to determine possible classes.
- (ii) The primary objects of the system are determined.
- (iii) Determining how each of the classes or objects are related to one another.
- (iv) All the appropriate relationships like generalization, association, etc. along with the proper naming must be included between each and every classes.
- (v) Finally with all the classes and appropriate relationships between them, the class diagram is created.

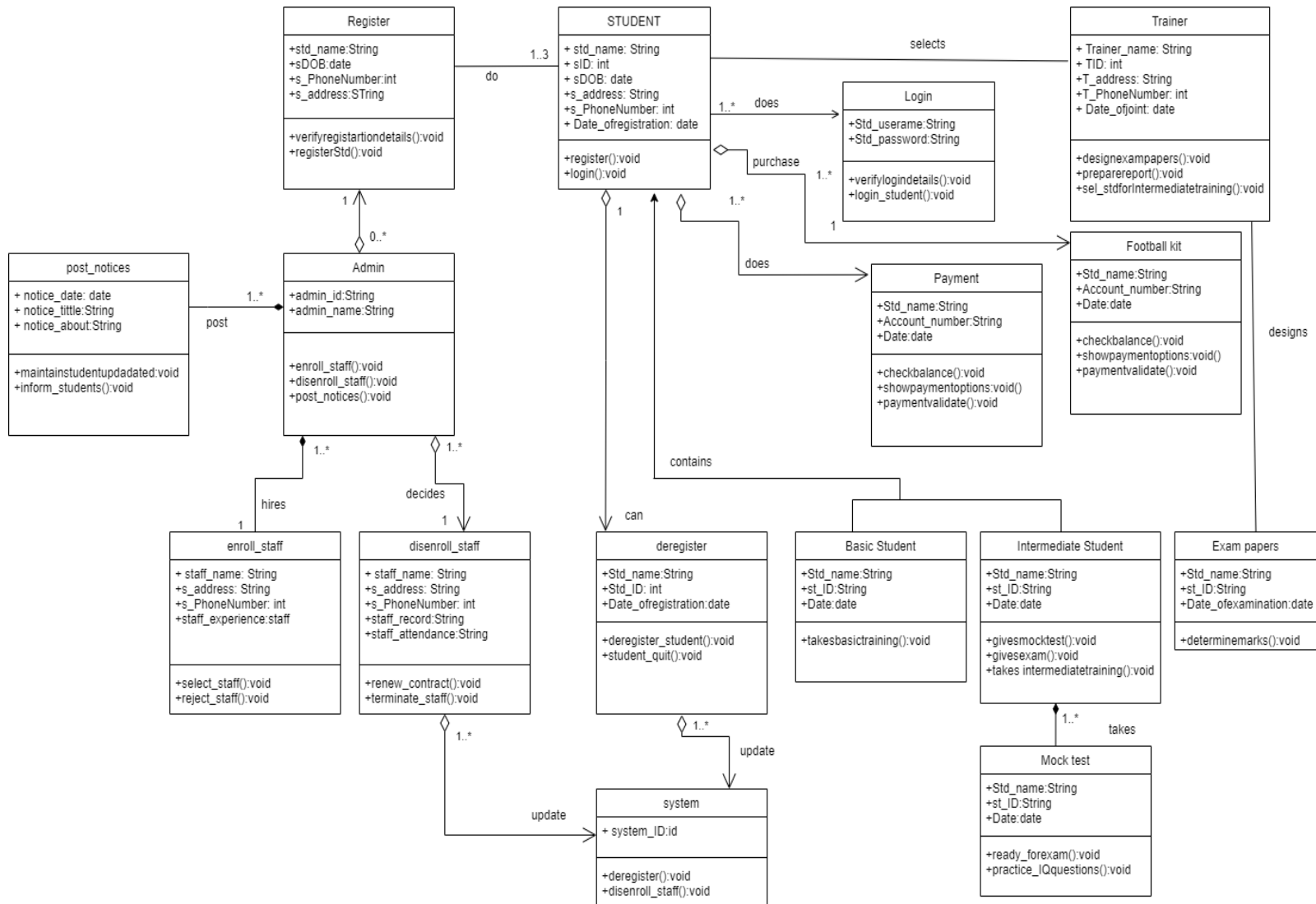


Figure 5: Class diagram

5. Further Development

This coursework' major goal was to create an online system for T14 Academy. Our system included a Gant chart, use case model, sequence diagram, collaboration diagram, and prototypes. This class used the Scrum Methodology. Agile scrum methodology is an incremental development project management approach. Each iteration is divided into two to four-week sprints, with the objective of completing the most critical features first and delivering a possibly deliverable product. Business modeling, design and analysis, execution, testing, and implementation were the four separate phases of the development process.

The fives phases of scrum methodology are listed below:-

- (i) Initiation: Develop a project Vision, Select Scrum Master and Representative(s), Form Scrum Team, Develop Epic(s), Create Prioritized Product Backlog, and Conduct Release Preparation are all procedures that occur during this phase.
- (ii) Plan and Budget: Generate User Stories, Accept, Estimation, and Submit User Stories, Create Work, Estimate Tasks, and Create Sprint Backlog are all procedures in this phase that deal with planning and estimating tasks.
- (iii) Integrate: This phase is concerned with completing the tasks and actions necessary to complete a project's product. Creating multiple deliverables, holding Daily Standup Meetings, and grooming the Product Backlog at regular intervals are all part of these tasks.
- (iv) Review and Reflection: This phase is involved with assessing the deliverables and completed work, as well as considering ways to enhance project work procedures and techniques.

- (v) Launch: This phase focuses on providing the customer's Accepted Work packages as well as recognizing, documenting, and internalise the project's lessons learned.

The next stage for our system would be to choose a system architecture. Classical Greek and Roman architecture would be my choice. The term "classical architecture" refers to a style that was popular in ancient Greece and Rome. This type of architecture was based on the idea of using a blueprint to construct structures. The temple, an oblong enclosure or encircled by columns, are common examples of architectural style. The Parthenon complex in Athens and the Colosseum in Rome are two of the most well-known examples of ancient architecture.

As a design pattern, I'd go with Creational - The Singleton Design Pattern for my system. The Singleton Design Pattern is a Creational pattern in which just one instance of a class is created and only one global access point to that object is provided.

So, in this way I would carry out the further development process of my online system.

7. Prototype

A prototype is a specimen, model, or launch of a product that was made to test an idea or procedure. A prototype is generally used to test a new concept in order to increase analyst and system user reliability. It is the stage between the formalization of a concept and its judgment.

The prototypes for our online system are pasted below:-

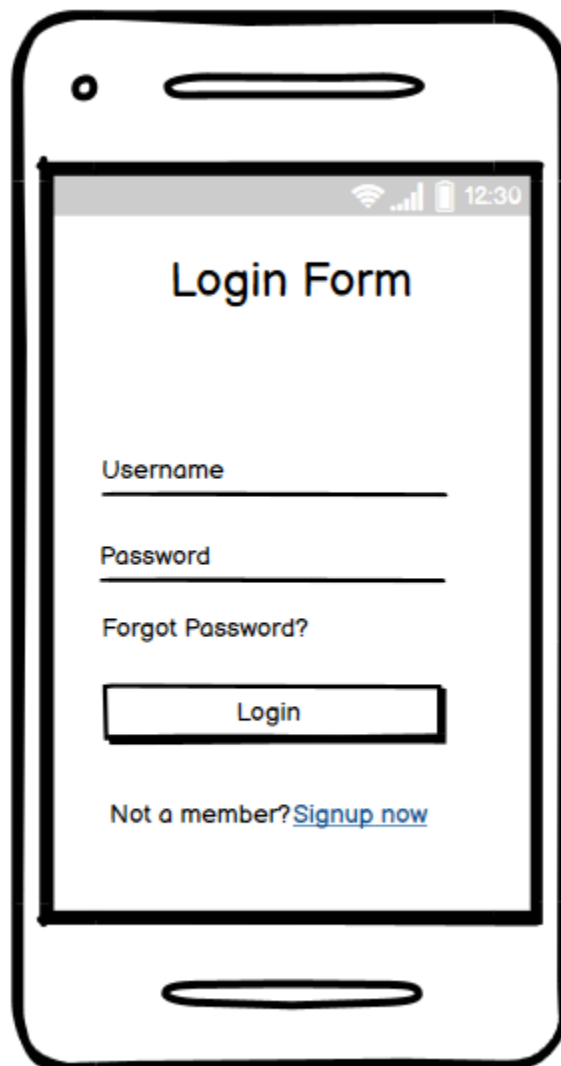


Figure 6: Prototype of Login Form

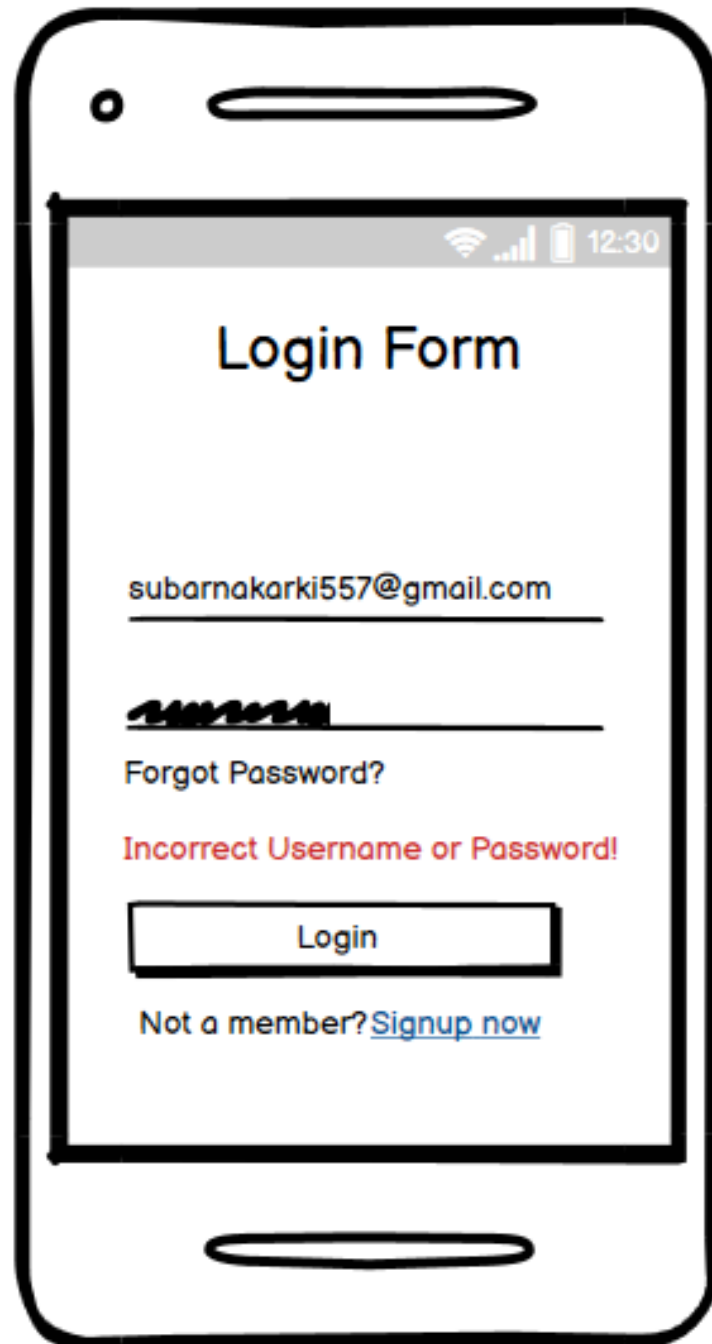


Figure 7: Prototype of Login Error

Registration Form

First Name Last Name

User Name

Email Address

Date of Birth

Gender

Password

Confirm Password

☐ I accept all the terms and conditions

Register

I am already Registered. [Login](#)

Figure 8: Prototype of Registration Form

Registration Form

First Name Last Name

User Name

Invalid Detail

Please enter valid details in all the fields

Ok

Register


I am already Registered. [Login](#)


Figure 9: Prototype of Registration Error


Payment Form



Choose payment Methods:

☐ Esewa ☐ Khalti ☒ FonePay

Amount 

FonePay id 

Utility Payment 

Account Number  

☒ Save Payment

Submit

Cancel

Figure 10: Prototype of Payment Form

The image shows a mobile application interface for a payment form. At the top, the status bar displays signal strength, Wi-Fi, and the time 13:47. The title 'Payment Form' is centered. Below it, the instruction 'Choose payment Methods:' is followed by three radio buttons: 'Esewa', 'Khalti', and 'FonePay' (which is selected). A text input field contains 'Rs.500' and a Nepali Rupee symbol icon. Below this field, a red error message reads 'Insufficient Balance, Try again later!'. The form continues with a phone number input field containing '09738737372' and a grid icon, a dropdown menu for 'Utility Payment', another phone number input field containing '11283847472' with an information icon, and a checked checkbox for 'Save Payment'. At the bottom are 'Submit' and 'Cancel' buttons.

Figure 11: Prototype of Payment Error

Payment Form

Transaction Success

Transaction Details

Reference Code	1223437
Date/Time	06 Apr 2002, 04:52 PM
Channel	Online
Payment Attribute	98849829282/Antivirus
Service Name	FONEPAY
Amount	2000.00
Initiator	98373617627

Done

Figure 12: Prototype of Payment success

14:42

Mock Test

search

Student ID

1) $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$?

☒ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

2) $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$?

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☒ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☐ $\frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2}$

☒ Confirm Submission

Submit

Cancel

Figure 13: Prototype of Mock Test

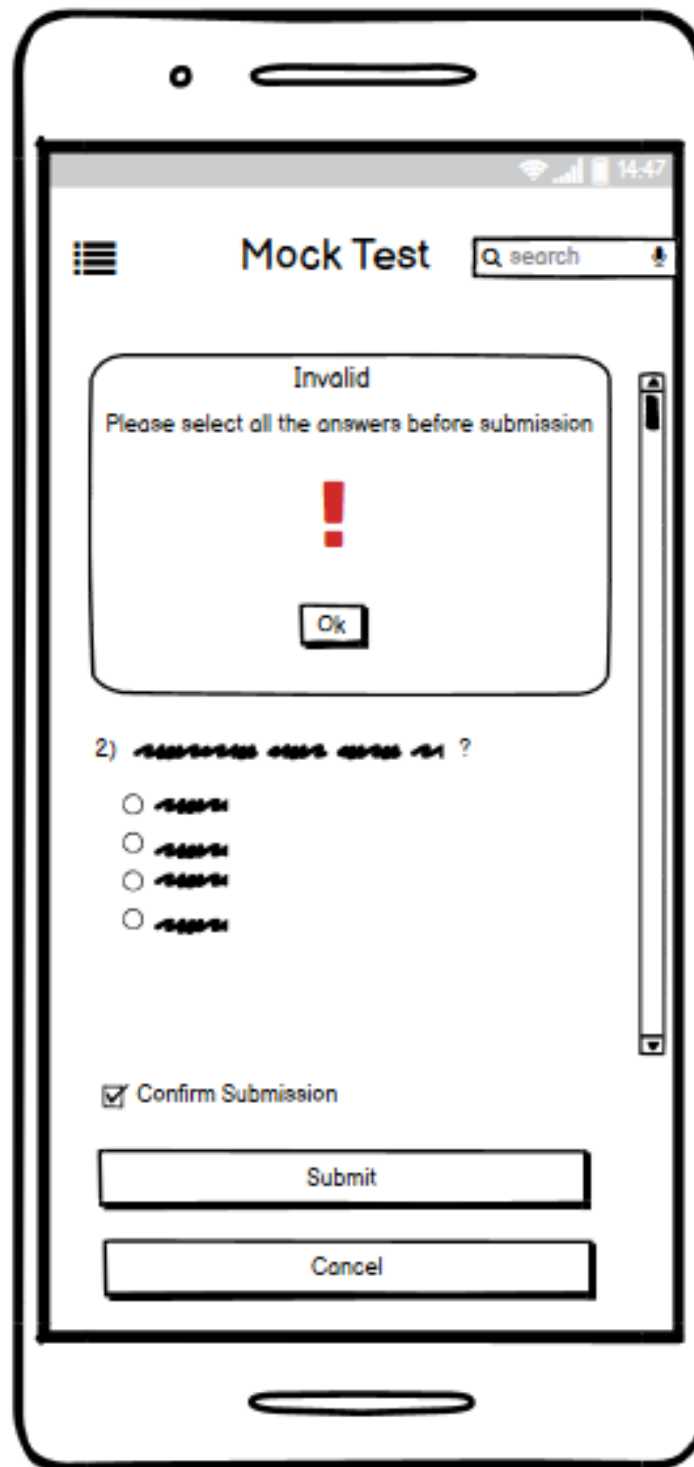


Figure 14: Prototype of Error in Mock test

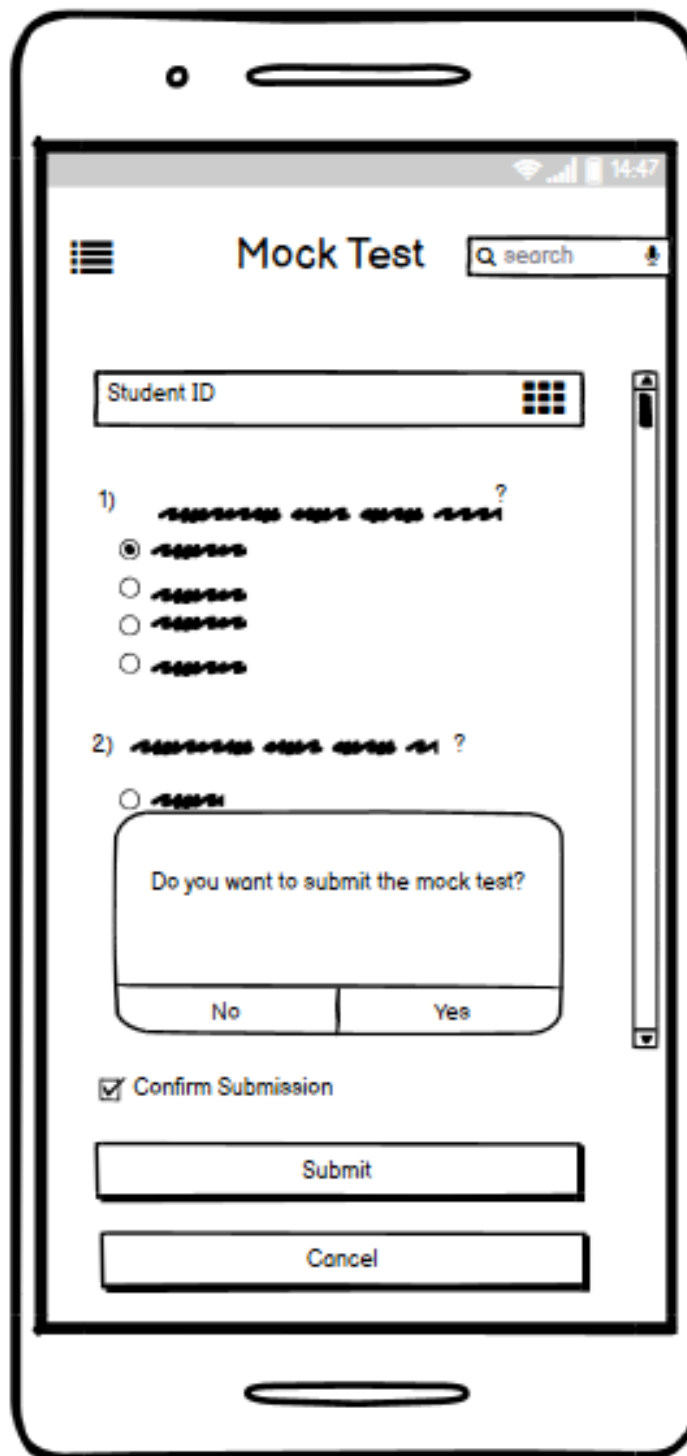


Figure 15: Prototype of Closing Mock Test



Figure 16: : Prototype of mock Test Submission

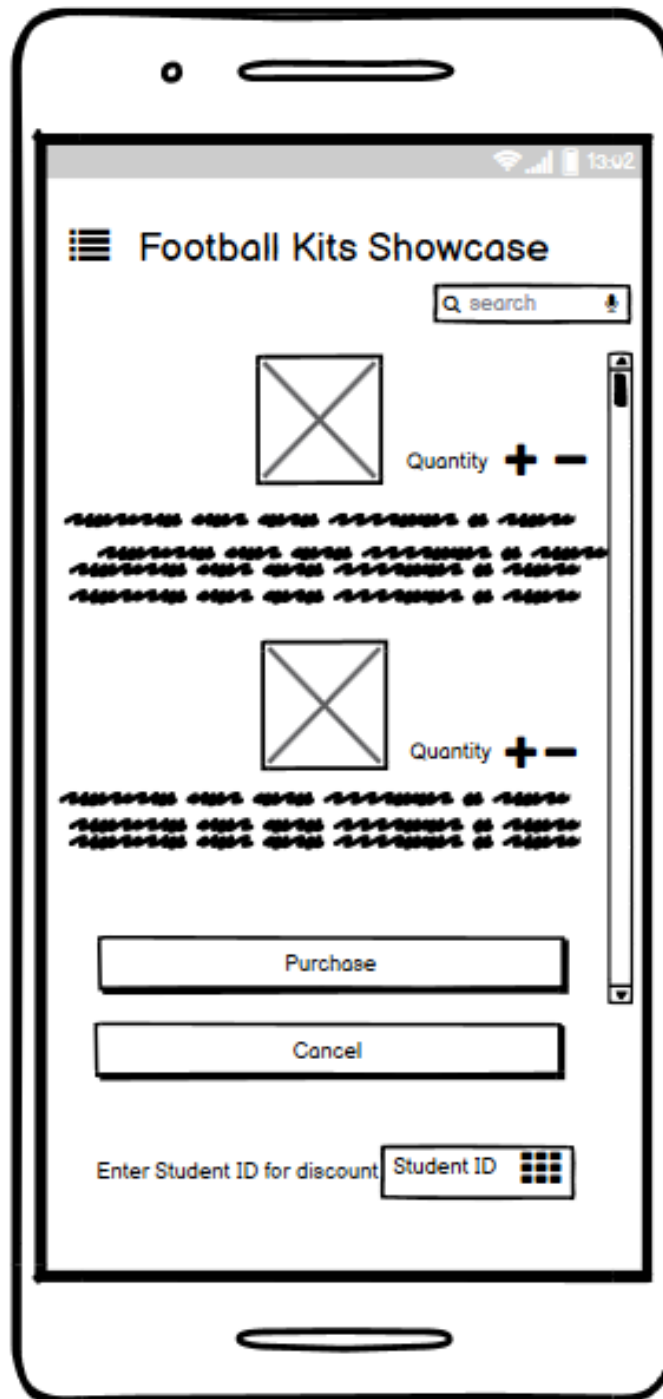


Figure 17: : Prototype of Football kit showcase

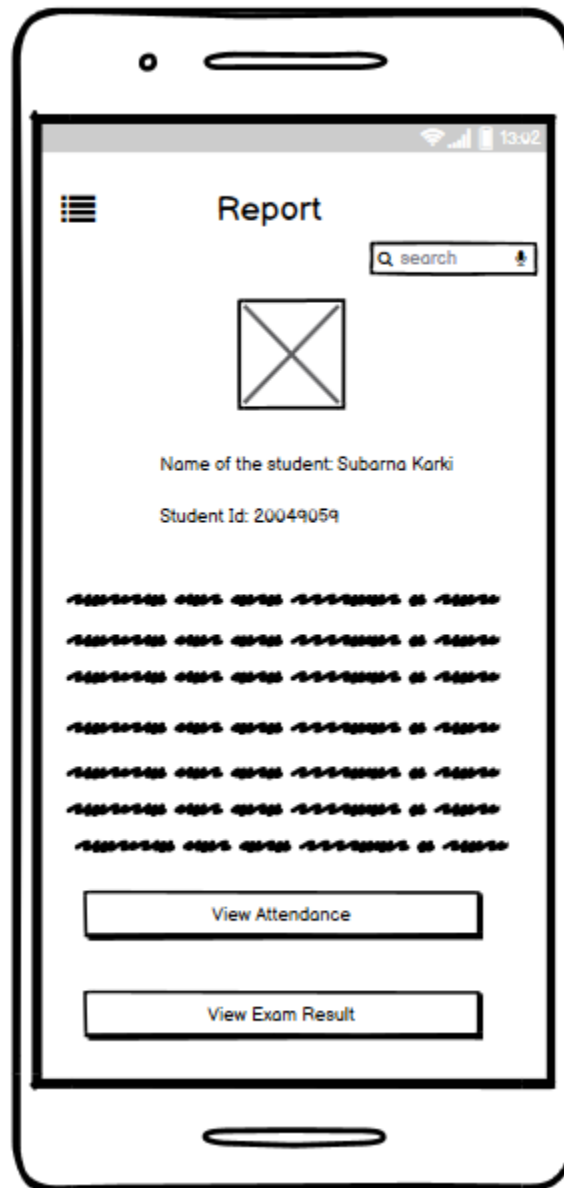


Figure 18: Prototype of report generation

8. Conclusion

To conclude, object oriented analysis and design was carried out for the creation of online system of T14 football academy. It was vital for the academy's business and revenue to grow. The design and analysis of the online system were completed by creating a Gantt chart to monitor the time duration, SCRUM methodology for the use case modelling, and the building of a prototype so that our system could be previsualized.

I ran across various issues and challenges while performing this task. To begin with, creating a Gantt chart for a system was extremely complicated and time intensive. It was pretty complex and confusing. The methodology was also a bit perplexing, and stuffing too many details into a single use case model added to the complexity. It was also a tremendous pain to utilize the right direction in between messages for the sequence and collaboration diagram.

I looked over the instructional lessons for that subject that our instructors had published on Google Classroom to clear up any confusions that arose while undertaking this assignment. Aside from Google Classroom, the library and the internet provided me with a wealth of books, diaries, and study materials. Most significantly, the teachers were extremely accommodating, answering all of my questions and clarifying any misunderstandings I had. Thus, I finished my coursework in this manner. The knowledge in this course was comprehensive and tedious since we were all new to software engineering and its different approaches and diagrams. In a nutshell, this course taught me the value of time and motivated me to work and accomplish my assignment in a short amount of time and under pressure.

Bibliography

investopedia. (2017, 11 12). *investopedia*. Retrieved from <https://www.investopedia.com/terms/g/gantt-chart.asp>

javatpoint. (2017, 09 23). Retrieved from <https://www.javatpoint.com/use-case-model>

lucidchart. (2018, 09 11). *lucidchart*. Retrieved from www.lucidchart.com:https://www.lucidchart.com/pages/uml-sequence-diagram

technologyevaluation. (2019, 09 11). *technologyevaluation*. Retrieved from www3.technologyevaluation.com:https://www3.technologyevaluation.com/solutions/53717/drawio

techopedia. (2012, 08 13). *techopedia*. Retrieved from [/www.techopedia.com:https://www.techopedia.com/definition/3818/inter-process-communication-ipc](http://www.techopedia.com:https://www.techopedia.com/definition/3818/inter-process-communication-ipc)

techtarget. (2019, 05 10). *techtarget*. Retrieved from www.techtarget.com:https://www.techtarget.com/searchsoftwarequality/definition/collaboration-diagram

tutorialspoint. (2020, 09 21). *tutorialspoint*. Retrieved from www.tutorialspoint.com:

visual-paradigm. (2018, 12 11). *visual-paradigm*. Retrieved from www.visual-paradigm.com:https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/

visualstudio. (2016, 11 21). *visualstudio*. Retrieved from www.visualstudio.com:https://code.visualstudio.com/docs