Abstract

The existing approach of excusing absences relied solely on email correspondence for managing absence excuses from lectures and exams and there is no doubt that this approach has its drawbacks in time-consuming and lack of visibility and tracking the status of absence excuse applications or knowing where they are in the review process .From this we come up with the idea of our project to transitioning from a solely email-based approach to a centralized platform that streamline the management of absence excuses.

This project aims to develop a platform where students can submit requests for excusing absences from lectures, mid-term exams, and final exams. The platform provides an interactive interface for students to electronically submit their absence excuses, allowing course instructors to review and process these requests efficiently. The automation introduced by this platform facilitates instructors in removing attendance records, reduce processing times, improve communication, and enhance overall efficiency in handling absence-related matters

Acknowledgement

All praise to ALLAH, Lords of the worlds and may the peace and blessings be on the most noble of the prophets, Muhammad (Peace Be Upon Him) and on his family and all his companions. We offer to ALLAH all praise and seek his help and forgiveness. We express our deepest gratitude to our parents, sisters and brothers. May Allah give them a long, happy and prosperous life. We would also like to thank all our sincere friends for their support and cooperation during this time span. we deem it our utmost pleasure to avail an opportunity to express our heartiest gratitude and deep sense of obligation to our honorable supervisor, Dr. Rashiq Marie for his dynamic and affectionate supervision, friendly behavior, inspiration and guidance throughout our thesis work. This work would have been remained a dream if he did not provide us all the support needed for the execution of our project. We express our deep sense of gratitude to our Dr. Muhannad Al Mohaimeed, Dr. Fahad Ghabban for they paid special attention with regard to my research work

Table of Content

List of Figures

List of Tables

List of Abbreviations

UML	Unified Modeling Language
ERD	Entity-Relationship Diagram
CI	Course instructor

Chapter 1: Introduction

1.1 Introduction

The problem of making excuses is a bit complicated, and to solve this problem, we came up with the idea of Ghiaby website. The purpose of Ghiaby website is to help students and facilitate the process and remove the excuses of absence electronically and quickly to remove the absence from the student and turn a process instead of e-mailing to a user-friendly website, and to the Vice-Deanship for Academic.

1.2 Problem Definition and Motivation

Ghiaby website is a website that helps students and faculty members regarding the issue of absence and converting excuses of absence from paper to electronic. The student submits an absence request on the website stating the reason for his absence from the lecture and uploading a copy of the excuse on the website. The request reaches the admin (Vice-Deanship for Academic) at the university, and a stamp is placed on it if the request is accepted or rejected. It's a kind of new idea for the university, and the purpose of the website is to help the students and others.

1.3 Project Objectives

The main objectives of transitioning from a solely email-based system to a centralized platform for managing absence excuses include:

- **Efficiency and Streamlining:** A centralized platform aims to streamline the process of submitting, reviewing, and approving absence excuses. It eliminates the delays inherent in email exchanges, allowing for quicker communication and resolution of absence-related issues.
- **The ability of tracking the application status:** The platform provides transparency by allowing both students and relevant faculty staff to track the progress of absence excuse requests. This transparency helps in understanding the status of applications and reduces ambiguity.

- **Automation and Notifications:** Implementing automated notifications within the platform helps in keeping course instructors and students informed about the status of absence excuse requests. This reduces the need for manual follow-ups and enhances the overall efficiency of the process.

1.4 Project Scope

This project is web-based application to helps students to make requests for deleting the absent by this process:

Student: Students login into the website. And make new requests on the request page. Upload the excuse with the reason why?

Admin: The request has been received by the admin (Vice-Deanship for Academic). The request is accepted or rejected. In case of rejecting: notification is sent to the student with note, why the excuse is rejected. In case of accepting: the student excuse reach to the management and its accepted. The excuse reaches to the course professor from whom the student is absent. The student will receive a notification of the completion of the order.

1.5 Project Timeline

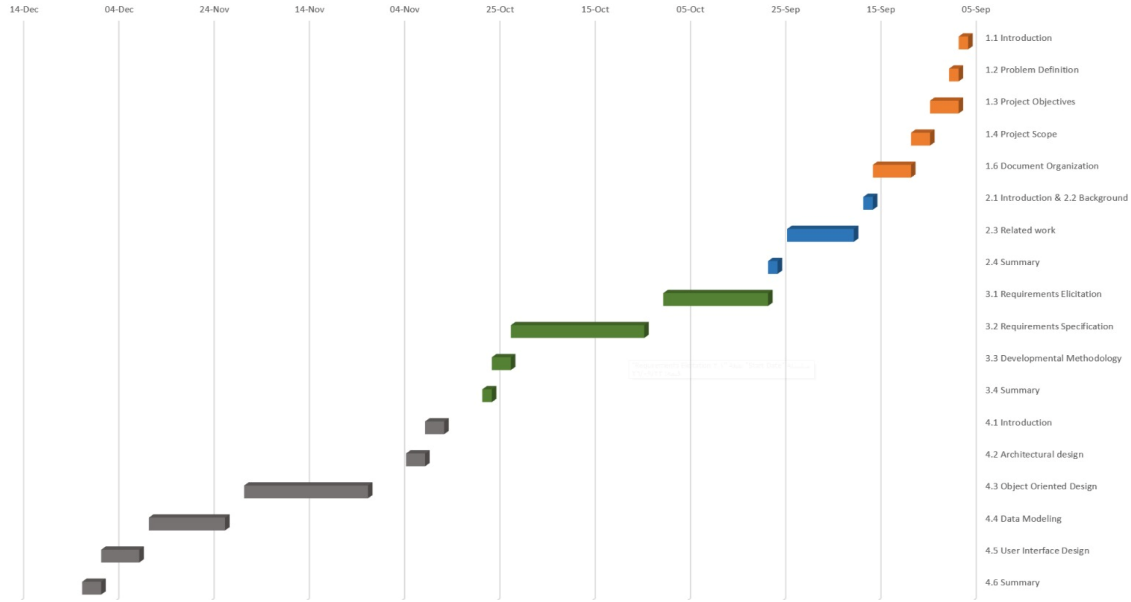

Figure 1.1 project timeline.

1.6 Document Organization

Chapter 1 Introduction: project motivation, objectives, scope and timeline.

Chapter 2 Literature Review: background and some related works.

Chapter 2: Literature Review

2.1 Introduction

Ghiaby website is website helps students and other staff to facilitate the excuse throw online process. We are trying to develop a fast and ease process to reduce the denial of attendance at lectures. The automated process will reduce the usage of paper and time and resources and help the university in the future.

2.2 Background

With the long process of deleting the absent from student and with E-mail correspondence the idea of Ghiaby website is presented. Ghiaby website is website to help students and facilitate the process and remove the excuses of absence electronically and quickly to remove the absence from the student.

2.3 Related work

2.3.1 Review of Relevant Projects

- **Qassim University:**

Request for an absent for attending the lecture for the subject professor.

Figure 2.1 Login Page (Qassim University)

The student has access to the "Explaining for absence of lectures" service through the following:

His account at the academic system portal and under the "students" tabulation will find the "excuses for absence of lectures" service and, through pressure on the service, two links appear to him:

One, "Introducing a request for excuses for absence of lectures" for a new request, two, "Follow-up to requests for excuses for absence of lectures" for follow-up requests.

Figure 2.2 Application Page (Qassim University) [1]

- **Jazan University :**

Absent Service.

Figure 2.3 Login Page (Jazan University)

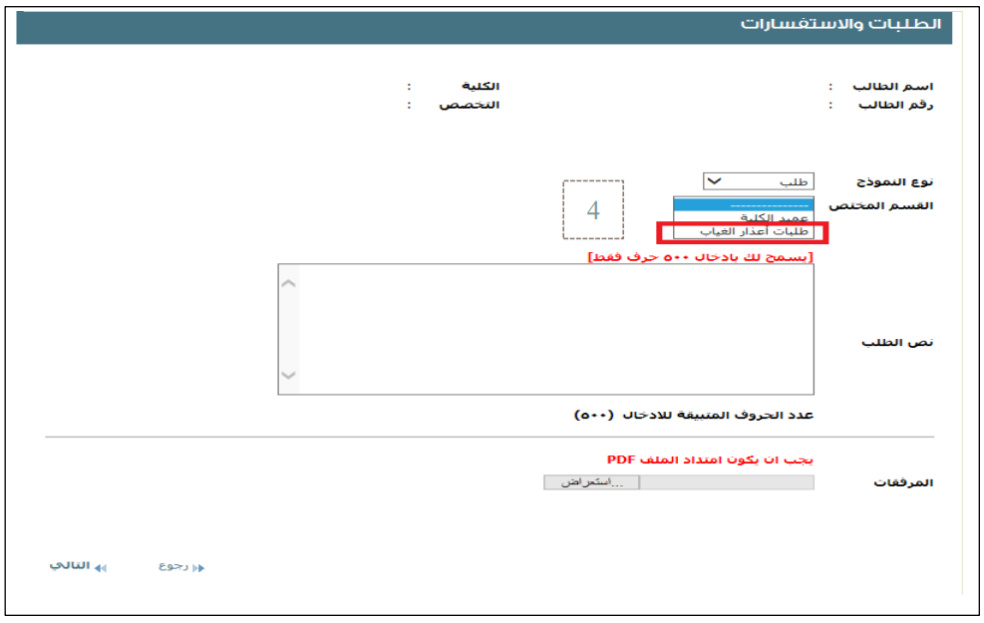

Figure 2.4 Application Page (Jazan University) [2]

- **Berkeley University :**

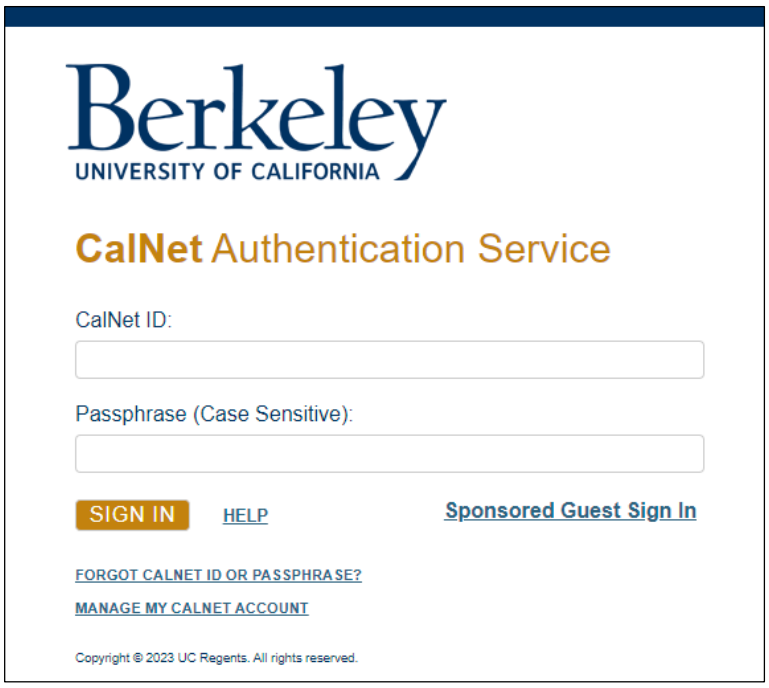

Figure 2.5 Login Page (Berkeley University)

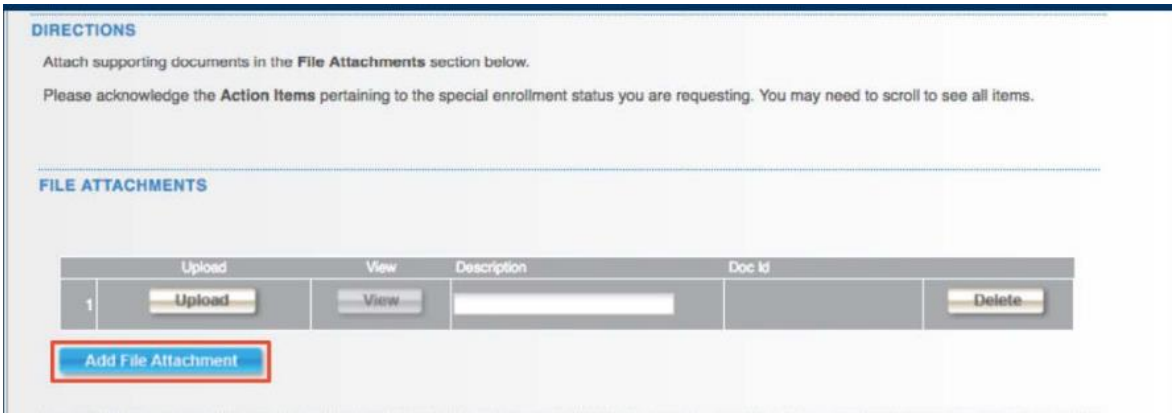

Figure 2.6 Application Page (Berkeley University) [3]

2.3.2 Relationship between the Relevant Work and Our Own Work

Features	(Request for an absent for attending) Qassim University	(Absent Service.) Jazan University	(CalNet center) Berkeley University	(Ghiaby website) Taibah University
Login	Yes	Yes	Yes	Yes
Home Page	Yes	Yes	Yes	Yes
New Application	Yes	Yes	No	Yes
Is the student able to check?	Yes	Yes	Yes	Yes
Is the platform for university students only?	Yes	Yes	Yes	Yes
Verification by the code?	No	No	No	Yes
Making excuses for midterm tests and quizzes?	No	No	Yes	Yes
Final tests?	No	No	No	Yes

Alarm when approaching (DN)	No	No	No	Yes
Parental Monitoring Sys?	No	No	No	Yes

Table 2.1 Comparison between Ghiaby website and other websites.

2.4 Summary

In this chapter, we introduced background information about Ghiaby website and a brief description of similar works. After comparing our proposed platform with these similar works, we conclude that our platform has unique features of its own.

Chapter 3: System Analysis

3.1 Requirements Elicitation

As students we face the problem of removal of absent and some students reach the forbidden percentage to get dined from taking the final exam and attending the classes.

The first question was about the students' gender, and whether they had problem for removal of their absent or not. According to the answers, we collected data from 55% of female students and 45% of male students.

The second question was whether the students they know where to go about the sent the absent. 59.6% of the students answered no, and 40.4% of the students answered yes. This is shown in the figure below see **(figure 3.1)**.

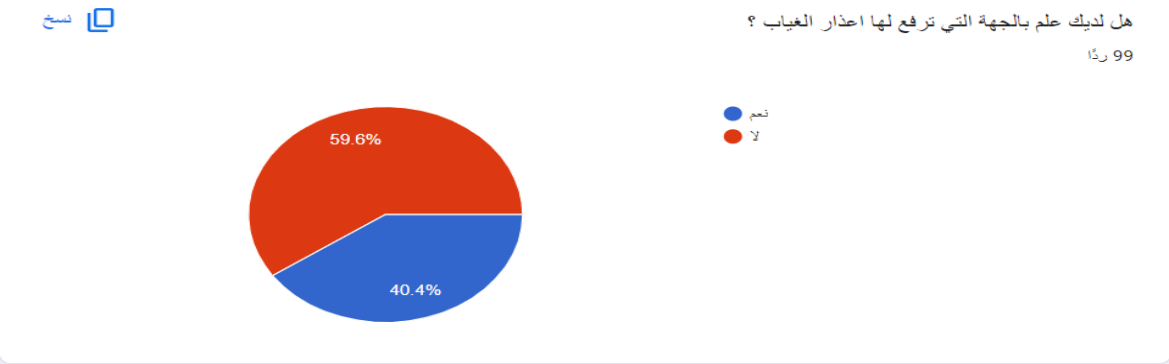

Figure 3.1: see if the student knew where to go about the absent

In the third question, we asked the student where they go if they have excuse. 78.8% put the first option, 12.1% put the second option, 15.2% put the third option, 30.3% put the fourth option. As shown in the figure below see **(Figure 3.2)**.

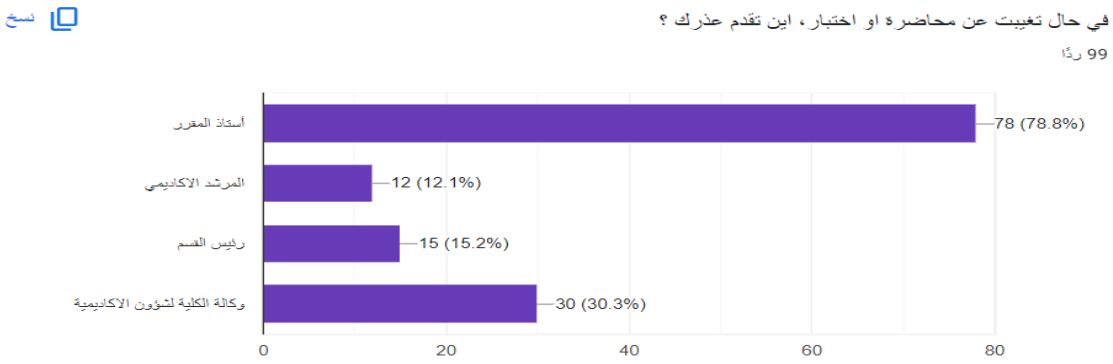

Figure 3.2: Question to the student to see where they present their excuse

In the fourth question, we asked the student about the maximum percentage allowed for the student before they denied attending the lectures if they know it or not. 84.7% of students answered yes, 15.3% of students answered no. As shown in the figure below see (Figure 3.3).

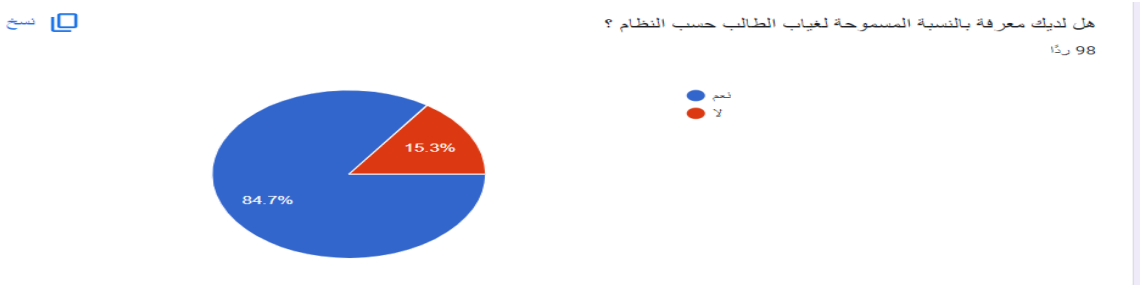

Figure 3.3: about the maximum percentage for absent

In the fifth question, we asked the students if they know where to send their absent excuse. 62.2% of students answered no, 37.8% answered yes. As shown on the figure below see **(Figure 3.3)**

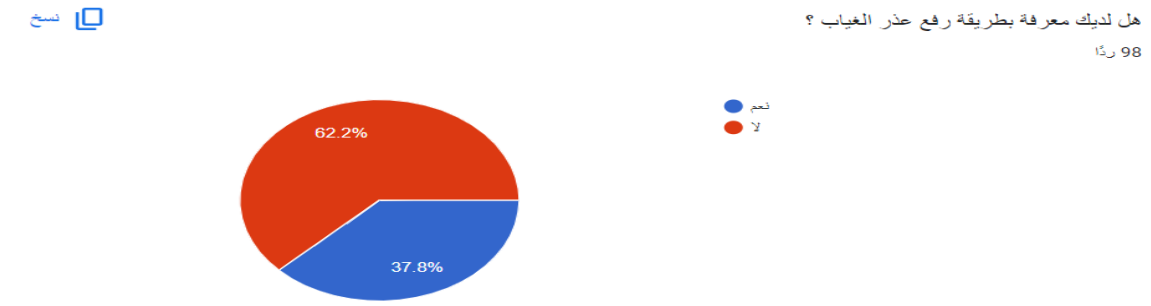

Figure 3.4: about how to send an absent excuse

In the sixth question, we asked the students if they know the maximum percentage exceeded without providing an acceptable excuse and will be forbidden from taking the final exam. 79.6% answered the 25% absent rate, 15.3% answered the 20% absent rate, 2% answered the 15% absent rate,3.1% answered the 30% absent rate. As shown in the figure below see **(Figure 3.5).**

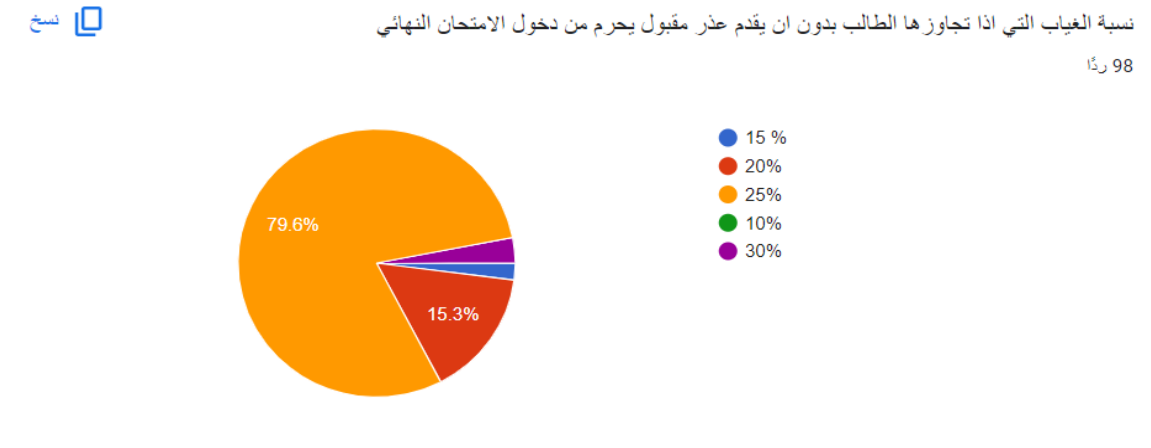

Figure 3.5: maximum percentage before the student is not allowed to attend the final exam

In the seventh question, we asked students about the current the absent excuse system is it easy and simple. 38.8% answered for “Neutral”, 26.6% answered for “I do not agree”, 22,4% answered for “Strongly disagree”, 8.2% answered for “Agree”, 4.1% answered for “Strongly agree”. As shown on the figure below see **(Figure 3.6)**.

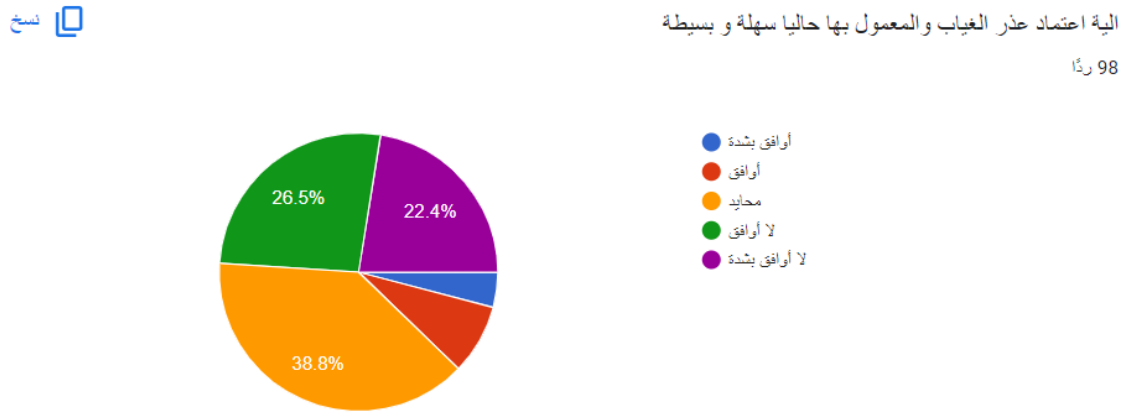

Figure 3.6: about the current absent excuse system

In the eighth question, we asked the students if they encountered any problems with the removal of the current absent they have. 73.7% encountered a problem and answered yes, 26.3% did not encounter any problem and answered no. As shown in the Figure below see **(Figure 3.7)**.

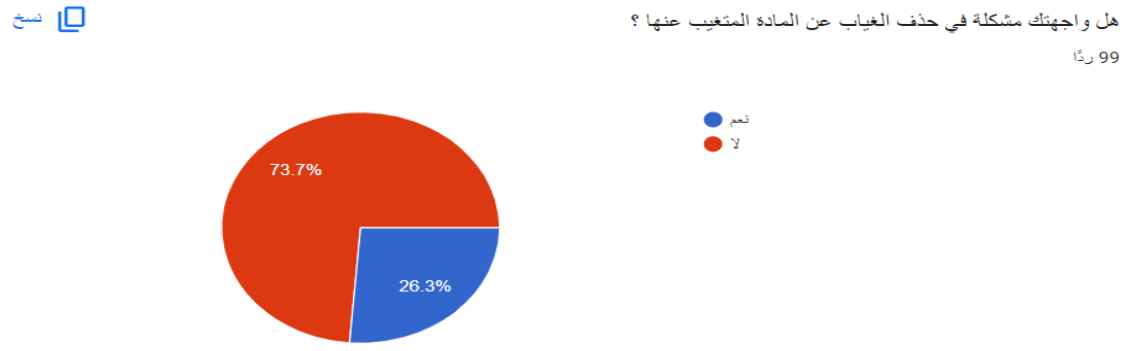

Figure 3.7: if the student face a problem with the removal of the absent

The figure below is for the previous chart. For the students answered yes.

Figure 3.7.1: for the “Yes” answered

The ninth question, we asked the students if they want to see a platform instead of the current absent excuse system. 74.7% answered for “Strongly agree”, 14.1% answered for “Agree”, 10.1% answered for “Neutral”, 1% answered for “Strongly disagree”. As shown in the figure below see **(Figure 3.8)**.

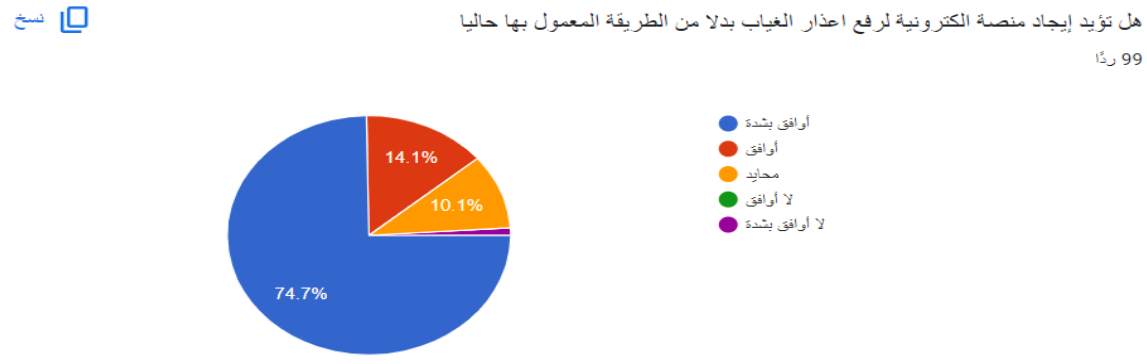

Figure 3.8: Number of those who agree or disagree the existence of the platform

In the tenth question, we asked the student to see if there any suggestions they wish to see in the. As shown in the figure below. See **(Figure 3.9)**.

Figure 3.9: Suggestions for the platform

3.1.1Functional Requirements

3.1.1.1 Requirements for Students:

- ▪ Sign-up page on the system.
- ▪ Log-in in the system
- ▪ can view the course schedule.
- ▪ can select the course he wants to rise application for absent
- .▪ can upload the excuse file.
- ▪ can view the application status.
- ▪ can manage their own profile.

3.1.1.2 Requirements for Admin:

- ▪ Sign-up on the system.
- ▪ Log-in in the system
- ▪ view received applications.
- ▪ view, accept, reject applications.
- ▪ manage their own profile.

3.1.1.3 Requirements for System:

- ▪ Send Email verification for successful registration to all users.
- ▪ Send Email notification for successfully submitted application.
- ▪ Send Email notification to students about the status of the applications.
- ▪ Send Email notification to Admin for new received applications.

3.1.2 Non-Functional Requirements

- ▪ **Availability**: The system is available all the time.
- ▪ **Compatibility**: The system is compatible with any device with a web browser.
- ▪ **Maintainability**: The system is easy to maintain and update.
- ▪ **Usability**: The user will not find any problem using the service.
- ▪ **Security**: The system must provide a high level of secure for personal information.

3.1.3 User Requirements or Domain requirements

3.1.3.1 User Requirements:

- ▪ Smart Device.
- ▪ Internet connection

3.2Requirements Specification

A use case diagram is one of the Unified Modeling Language (UML) diagrams that represents how the user interacts with the system. It describes actors and their functionalities in the system based on their roles.

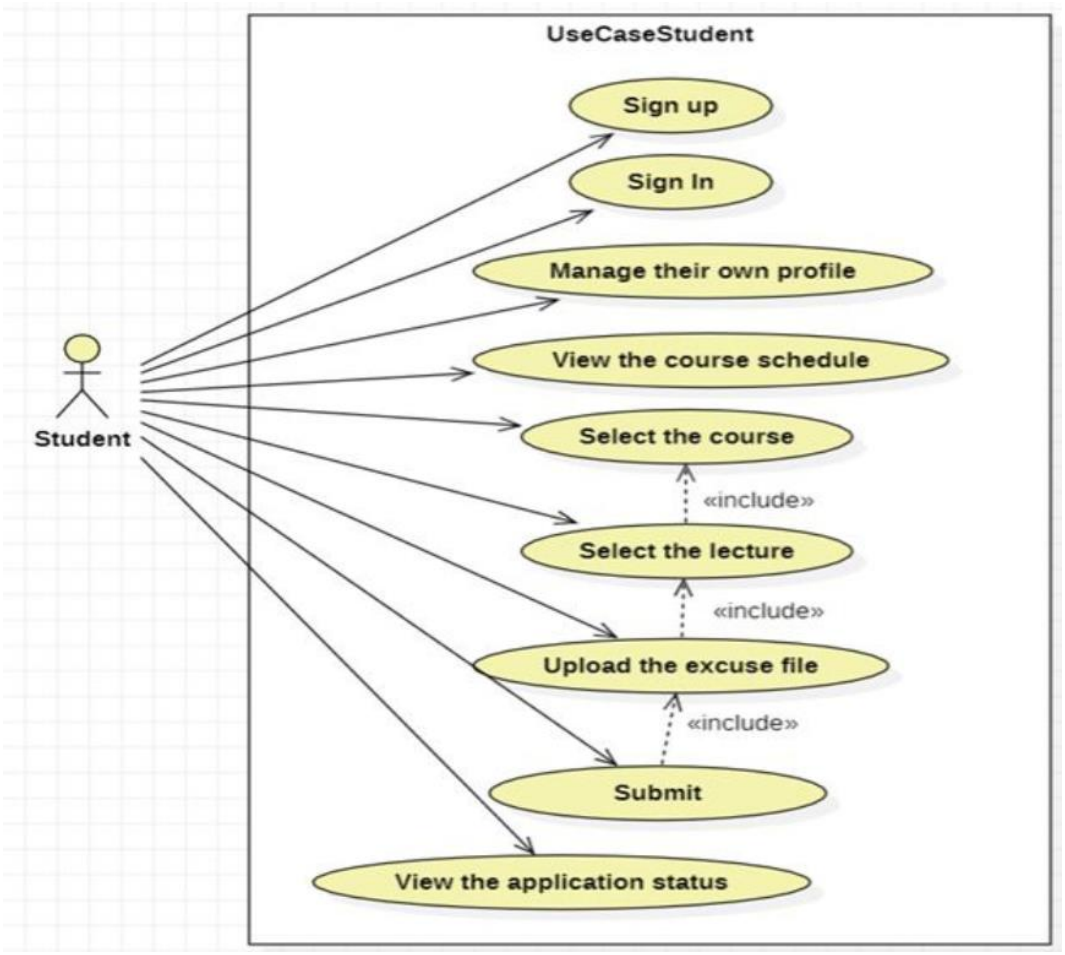

Figure 3.10: Ghiaby website Use case (Student).

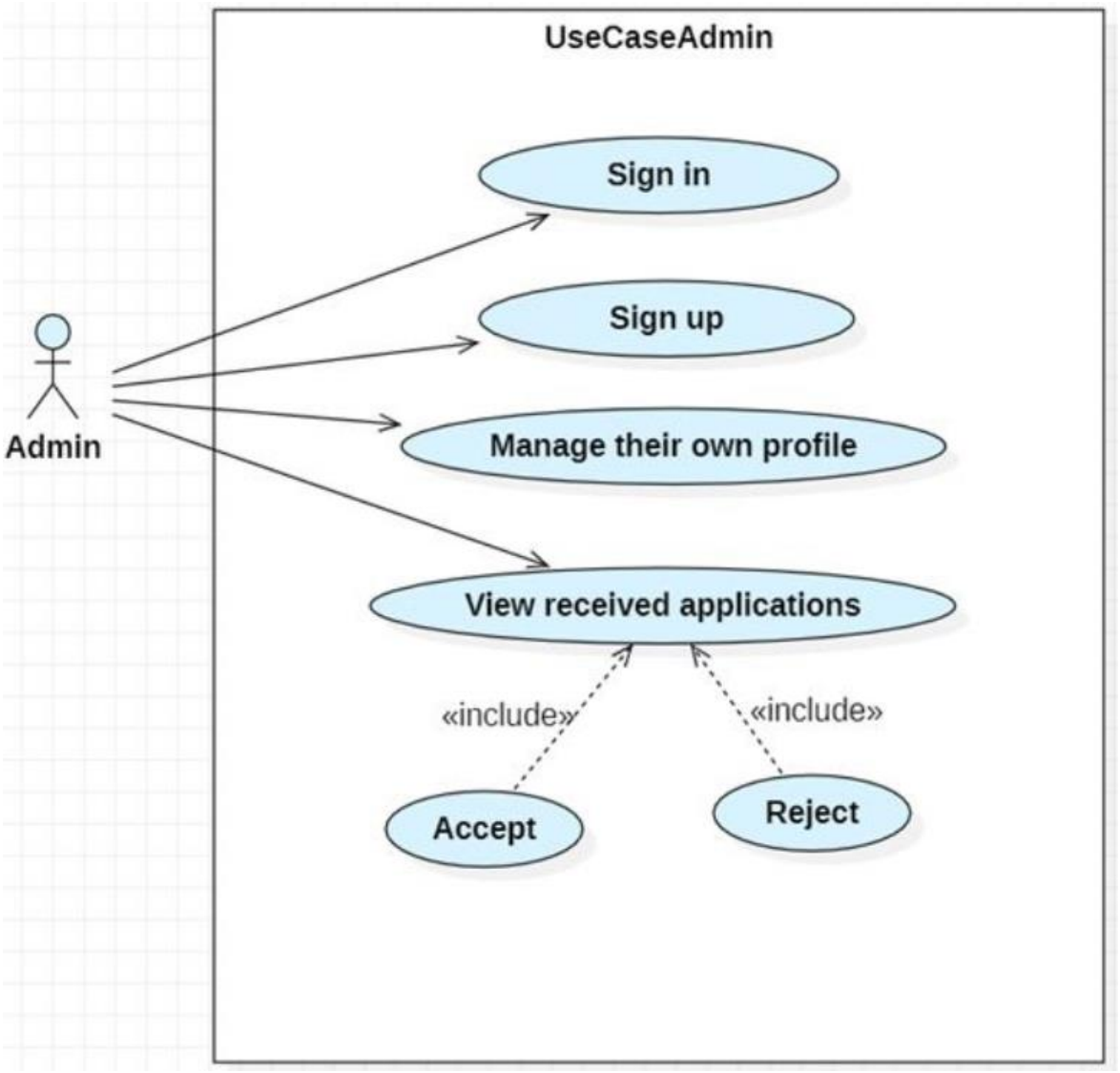

Figure 3.11: Use case (Admin).

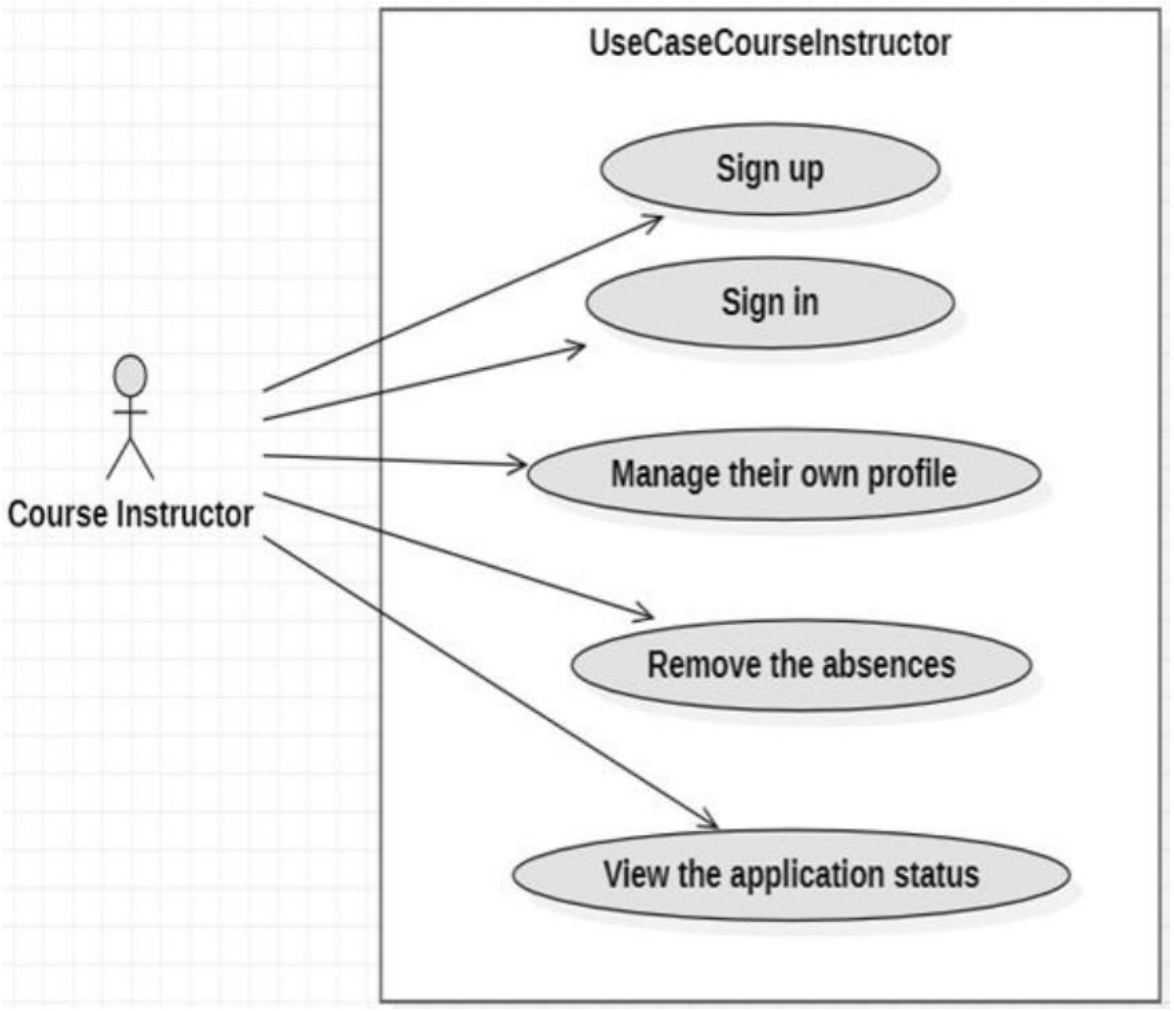

Figure 3.12: Use case (Course instructor).

Use Case ID [#001]	Sign up.
Actor	Students, Admin, Course Instructor.
Description	Unregistered student, admin and Course Instructor can enter their data to create their own account.
Precondition	Unregistered student, admin and Course Instructor must not have previously registered in the system and enter valid data.
Basic flow	1. Click on sign-up. 2. The user enters all necessary data. 3. Click register.
Exception	▪ Invalid credentials. ▪ Credentials already used.

Table 3.1: Sign up use case description.

Use Case ID [#002]	Sign in.
Actor	Students, Admin, Course Instructor.
Description	student, admin, and Course Instructor enter their credentials to login to the system.
Precondition	student, admin, and Course Instructor must be registered in the system.
Basic flow	1. Click on sign-in. 2. Students, admin and Course Instructor enter their username and password. 3. If credentials are correct, then each one of them will be redirected to the home page that corresponds to his role.
Exception	▪ Account does not exist. ▪ Invalid credentials.

Table 3.2: Sign in use case description.

Use Case ID [#003]	Manage their own profile.
Actor	Students, Admin, Course Instructor.
Description	student, admin, and Course Instructor can add, delete and modify their data to their own.
Precondition	Sign in.
Basic flow	1. Click on sign-in. 2. Click profile. 3. Can edit the profile.
Exception	▪ None.

Table 3.3: Manage their own profile description.

Use Case ID [#004]	View the course schedule.
Actor	Student.
Description	The student can view the schedule of subjects to do what he wants
Precondition	Sign in.
Basic flow	1. Click on sign-in. 2. Click View the course schedule.
Exception	▪ None.

Table 3.4: View the course schedule.

Use Case ID [#005]	Select the course.
Actor	Student.
Description	The student can choose the course that he previously missed and upload the excuse.
Precondition	Sign in.
Basic flow	1. Click on sign-in. 2. Click View the course schedule. 3. Can Select the course.
Exception	▪ None.

Table 3.5: Select the course.

Use Case ID [#006]	Select the lecture.
Actor	Student.
Description	Students can select lecture date that he missed and received an absent.
Precondition	Select the course.
Basic flow	1. Click on sign-in. 2. Click to view the course schedule. 3. Can select the course. 4. Select the lecture.
Exception	▪ None.

Table 3.6: Select the lecture.

Use Case ID [#007]	Upload the excuse file
Actor	Student.
Description	The student can file a request in the Submit Excuse box.
Precondition	Select the course.
Basic flow	1. Click on sign-in. 2. Click to view the course schedule. 3. Can select the course. 4. Can upload the excuse file.
Exception	▪ None.

Table 3.7: Upload the excuse file.

Use Case ID [#008]	View the application status.
Actor	Student.
Description	The student can know the status of the application by viewing the application status.
Precondition	Sign in.
Basic flow	1. Click on sign-in. 2. View the application status.
Exception	▪ None.

Table 3.8: View the application status.

Use Case ID [#009]	View received applications
Actor	Admin.
Description	The admin can view received to be accepted or rejected.
Precondition	Sign in
Basic flow	1. Click on sign-in. 2. Click view received and select the applications. 3. Click on "Accept" or "Reject". o If “Accept” is clicked, the application is forwarded to the course instructor. o If “Reject” is clicked, the student’s application status will be changed.
Exception	▪ None.

Table 3.9: View received applications.

Use Case ID [#0010]	Remove the absences.
Actor	Course instructor.
Description	Course instructor can view the received applications in his system and based on the admin decision he remove the students absent.
Precondition	Sign in
Basic flow	1. Click on sign-in. 2. View the received application. 3. Click to Remove the absences.
Exception	▪ None.

Table 3.10: Remove the absences.

3.3 Developmental Methodology

Developmental Methodology: There are many developmental methodologies used in projects, the Agile methodology being one of the more prominent ones.” Agile” is an approach to software development that seeks the continuous delivery of working software created in rapid iterations. We picked the agile development approach because it matches our project vision, as we are working simultaneously over short period of time.

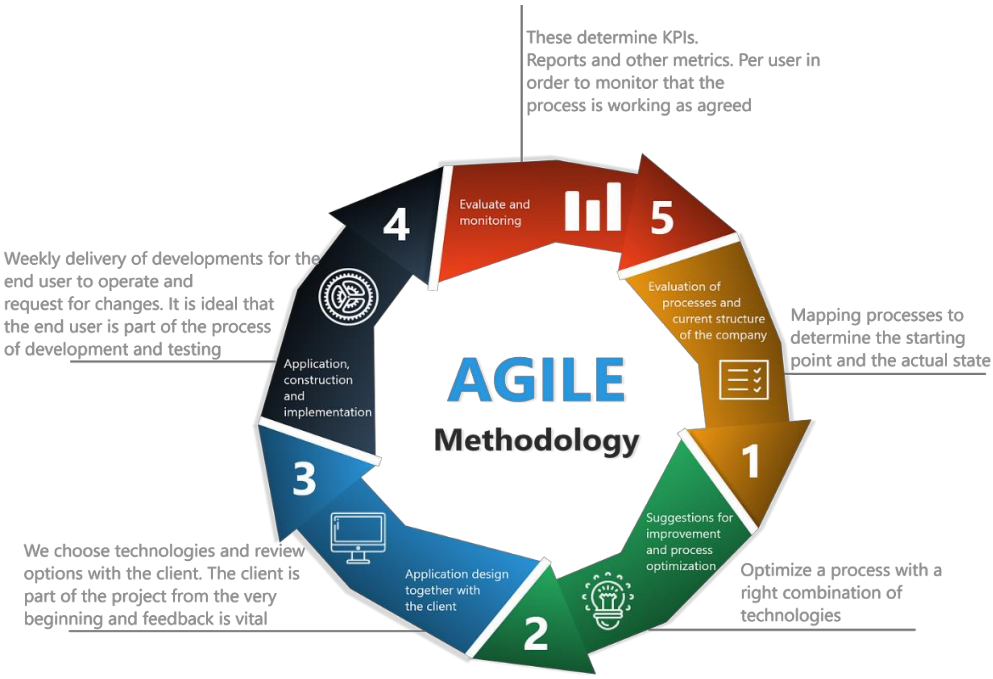

Figure 3.13: Agile methodology

3.4 Summary

In this chapter, we showed the result of our questionnaire. We discussed the functional and non-functional requirements. After that we showed use-case diagrams, with a detailed description of every use case. Lastly, we talked about the development.

Chapter 4: System Design

4.1 Introduction

In this chapter, we will represent the system design as architectural design, and as object-oriented designs that include both structural static models and dynamic models. For the structural static models, we will show the class diagram and the ERD. As for the dynamic models we will show several sequence diagrams. And lastly, we will show the data modeling.

4.2 Architectural design

The architectural design is concerned with understanding how a system should be organized and designing the overall structure of that system, i.e. it involves identifying major system components and their communications.

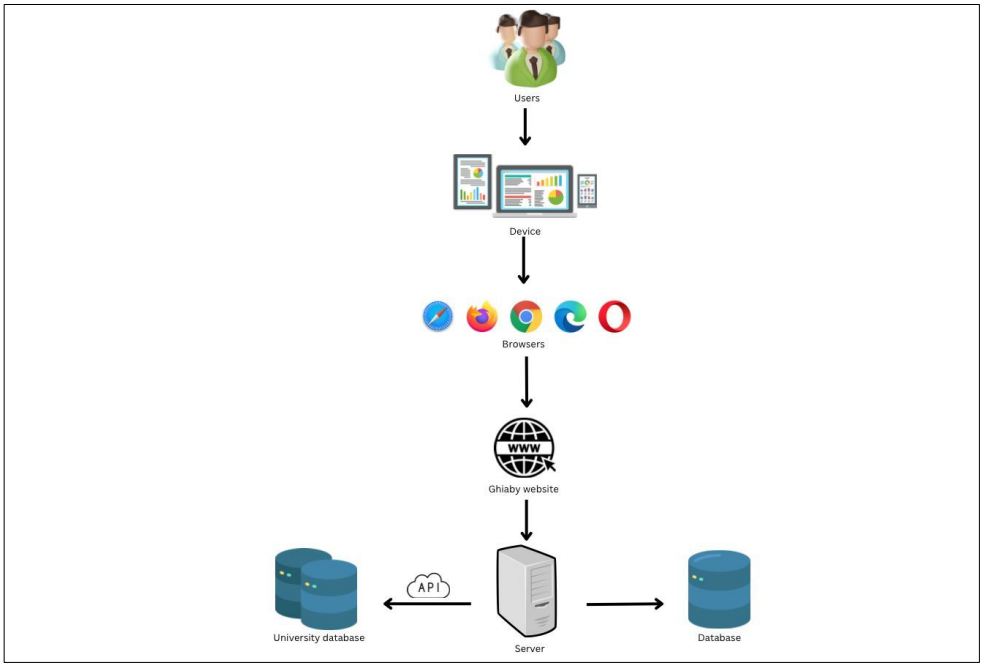

Figure 4.1: Architectural design.

4.3 Object Oriented Design

4.3.1 Structural Static Models

4.3.1.1 Entity-Relationship Diagram

is a structural design for databases. It shows visual representation for the entities with its attributes and the relationship between these entities. This is shown in the following **(figure 4.2).**

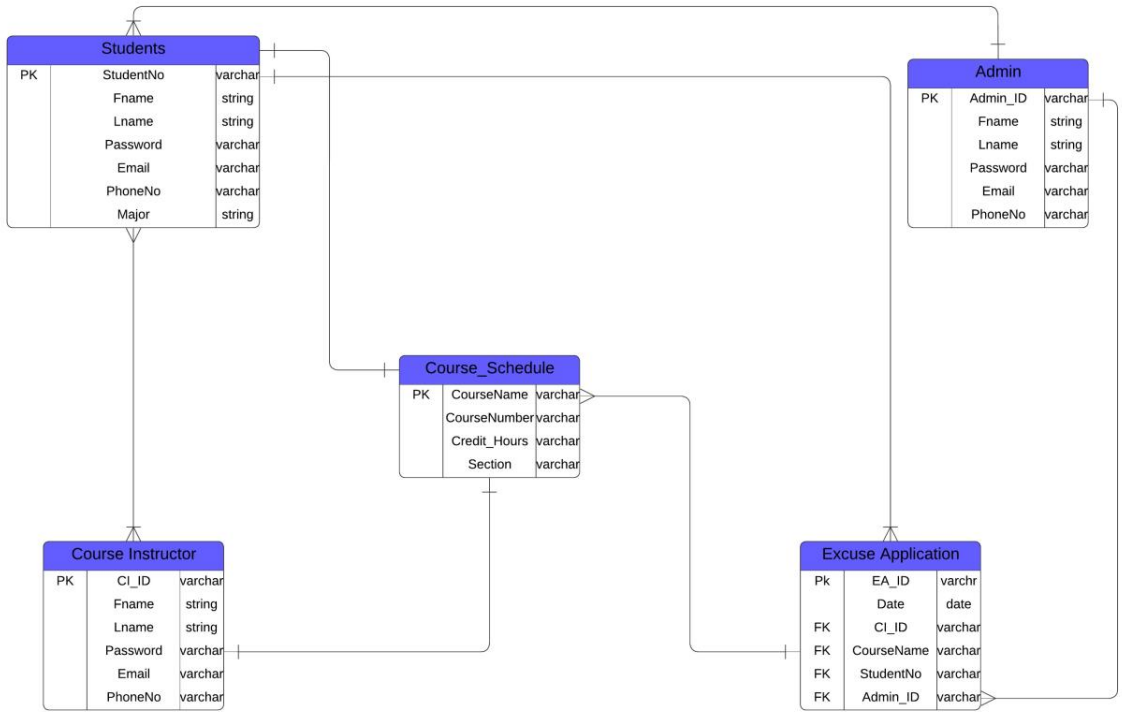

Figure 4.2: ER Diagram

4.3.1.2 Class Diagram:

A class diagram is a static view, and it is one of the UML diagrams that represent the classes and their relationship with each other. Each class has its own attributes and methods The class diagram is shown in (figure 4.3) below.

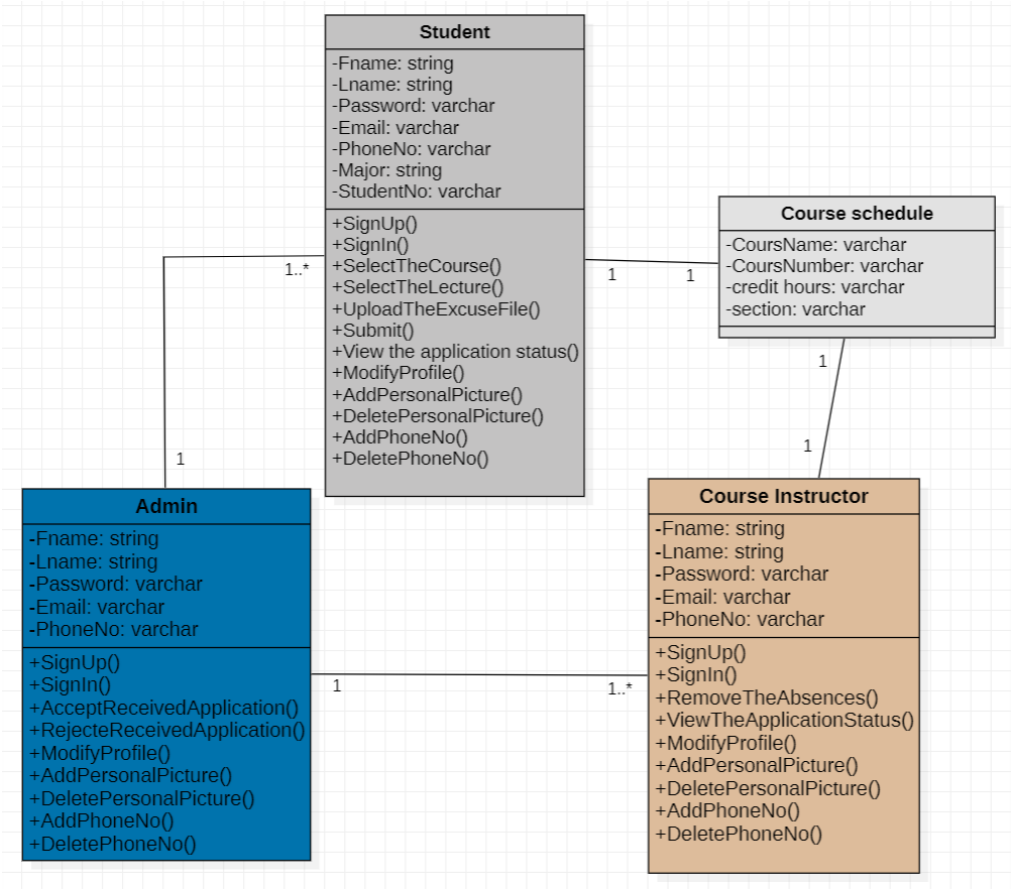

Figure 4.3: Class Diagram

4.3.2 **Dynamic Models**

A sequence diagram is a type of diagram that is used in UML diagrams to represent the interactions between objects in sequential order. It is used to describe the flow of messages, events, and actions between objects over time. This is shown in figure 4.4 through figure 4.7.

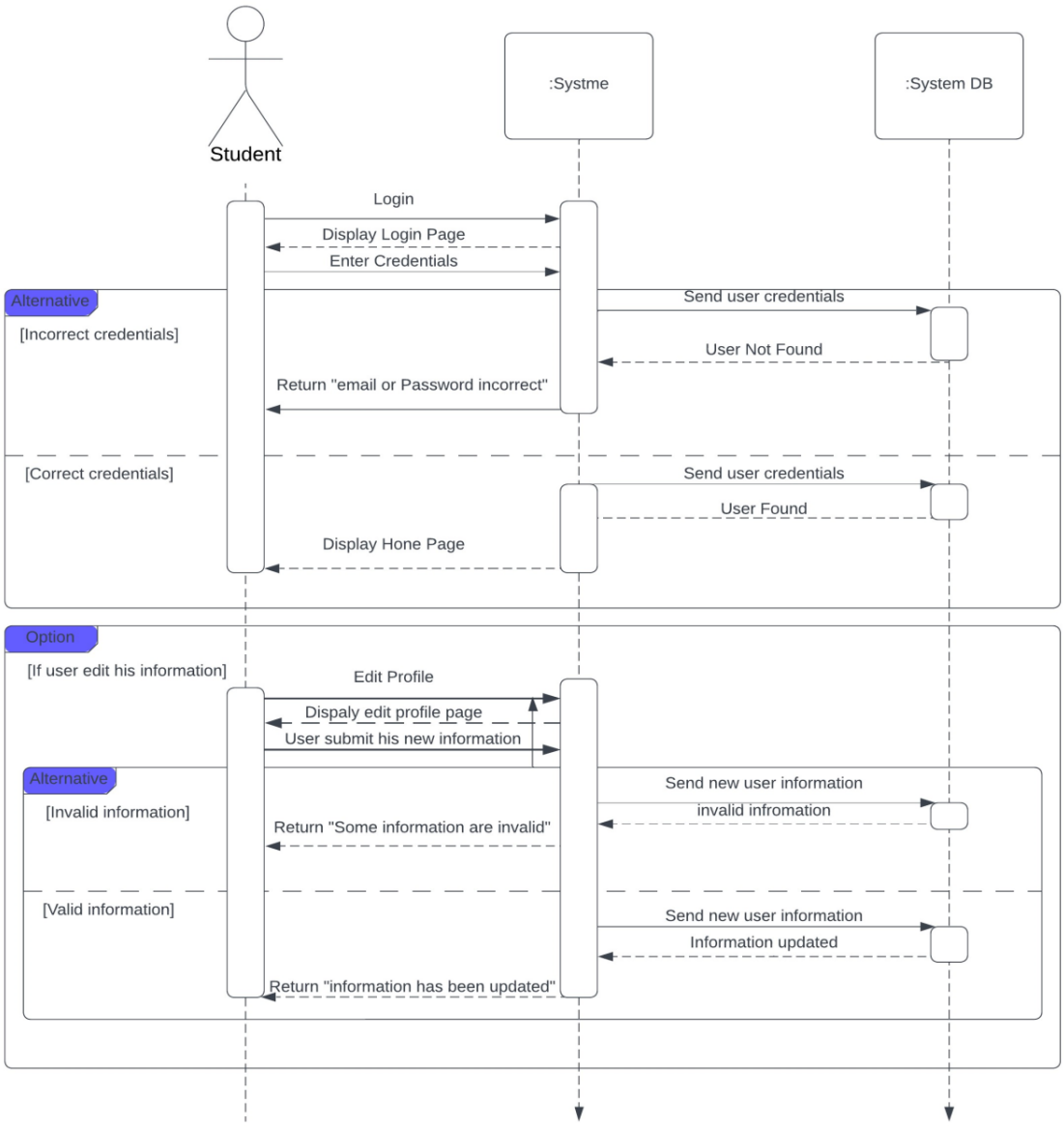

Figure 4.4: Login Sequence Diagram.

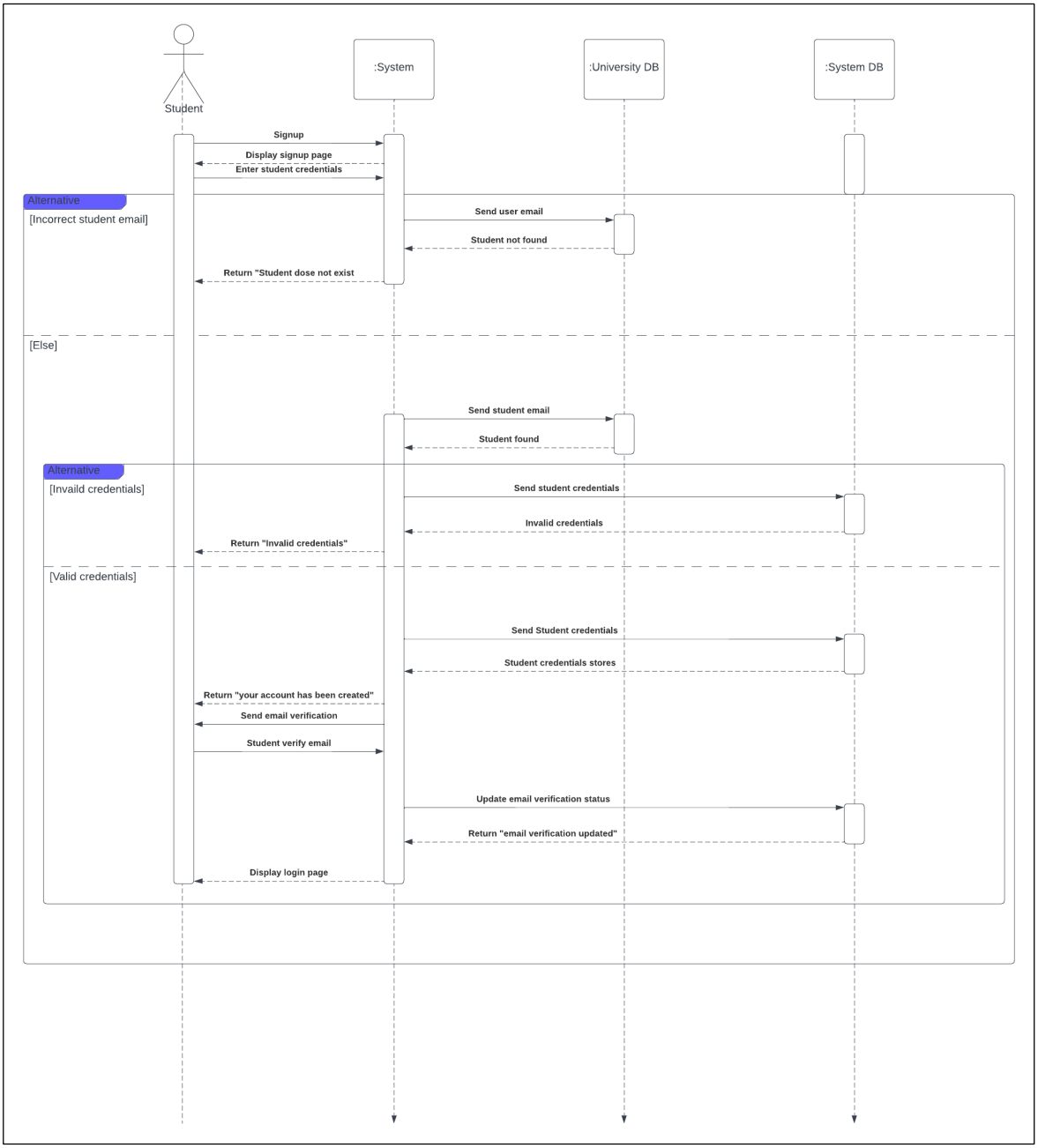

Figure 4.5: Signup Sequence Diagram.

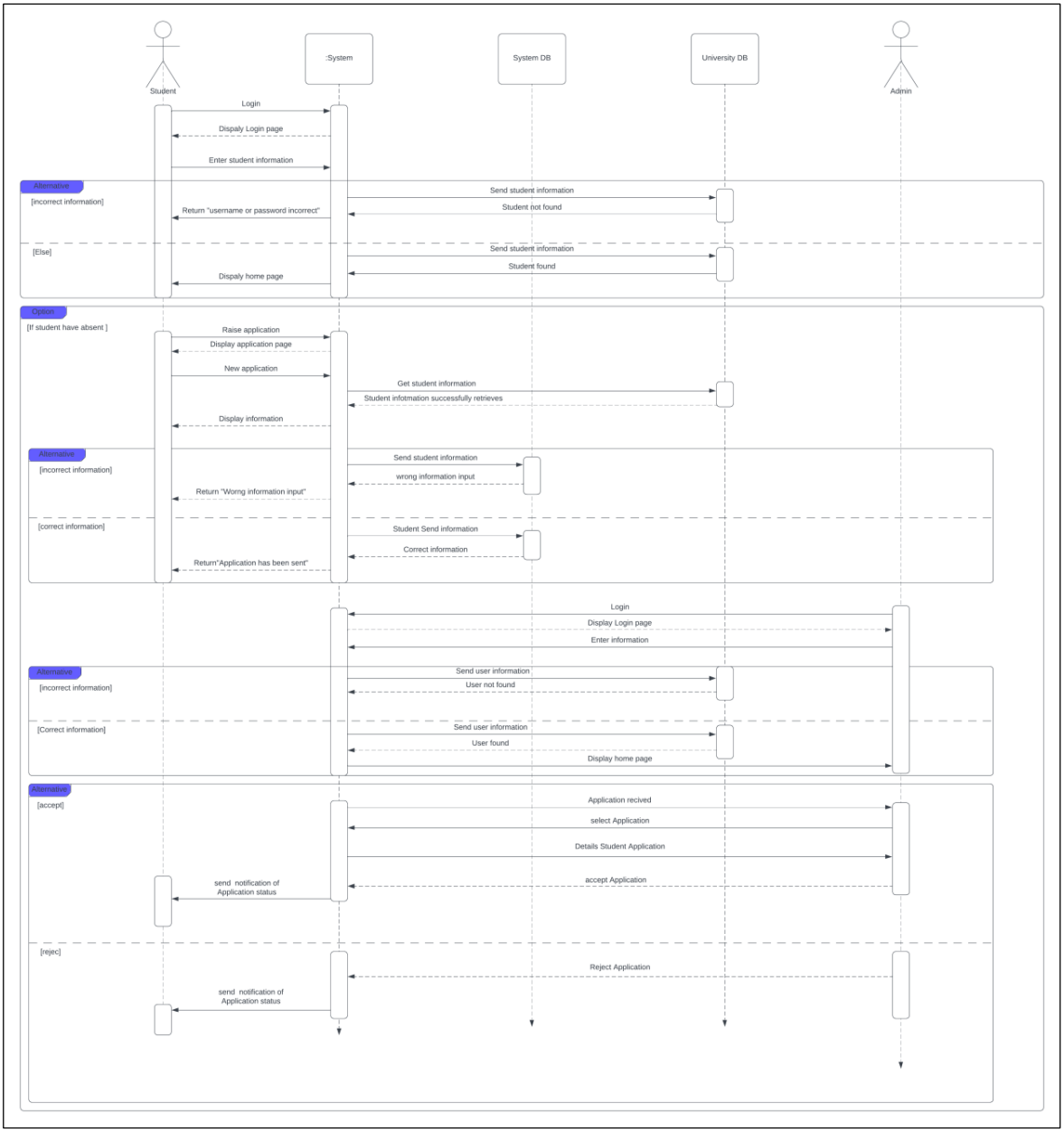

Figure 4.6: Raise Application Sequence Diagram

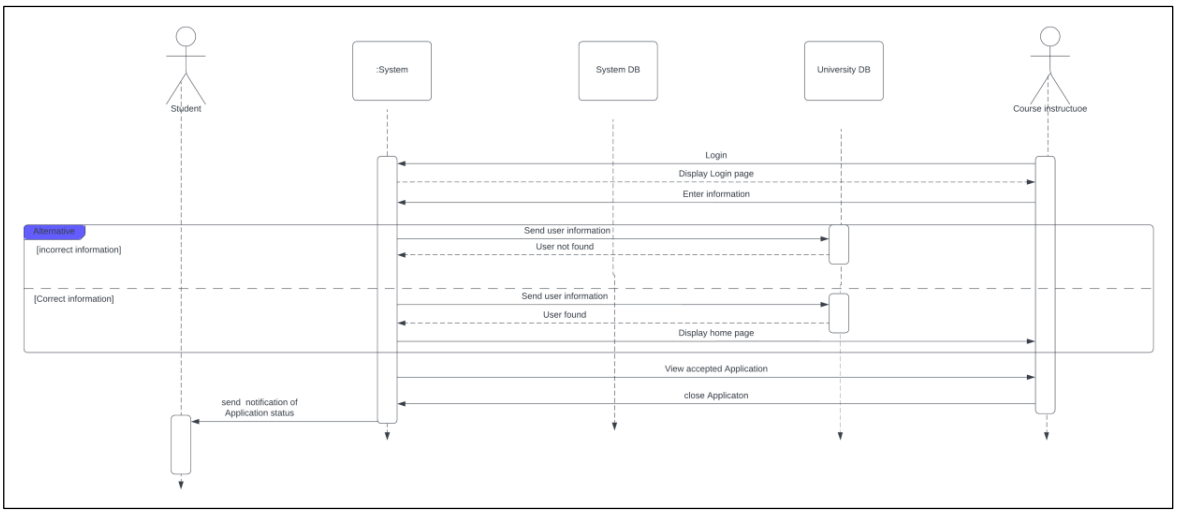

Figure 4.7: Submet Application Sequence Diagram

4.4 Data Modeling

In this section, we will represent tables that explain entities' attributes in ERD. We will show descriptions of the attributes, their types, and their constraints in the database. This is shown in table 4.1 to table 4.5.

Attribute	Description	Type	Primary Key (PK)	Foreign Key (FK)
StudentNo	Student number	VARCHAR	PK	
Fname	Student First name	String		
Lname	Student Last name	String		
Email	Student Email	VARCHAR		
Password	Student password	VARCHAR		
PhoneNo	Student Phone number	VARCHAR		
Major	Student Major	String		

Table 4.1: Student Table

Attribute	Description	Type	Primary Key (PK)	Foreign Key (FK)
AdminId	Admin number	VARCHAR	PK	
Fname	Admin First name	String		
Lname	Admin Last name	String		
Email	Admin Email	VARCHAR		
Password	Admin Password	VARCHAR		
PhoneNo	Admin Phone number	VARCHAR		

Table 4.2: Admin Table

Attribute	Description	Type	Primary Key (PK)	Foreign Key (FK)
Cnumber	Course Name	VARCHAR	PK	
Cname	Course Number	VARCHAR		
CreditH	Course Credit Hours	VARCHAR		
Section	Course Section	VARCHAR		

Table 4.3: Course Schedule Table

Attribute	Description	Type	Primary Key (PK)	Foreign Key (FK)
CIid	Course Instructor Id	VARCHAR	PK	
Fname	Course Instructor First name	String		
Lname	Course Instructor Last name	String		
Email	Course Instructor Email	VARCHAR		
Password	Course Instructor password	VARCHAR		
PhoneNo	Student Phone number	VARCHAR		

Table 4.4: Course Instructor Table

Attribute	Description	Type	Primary Key (PK)	Foreign Key (FK)
AppId	Application number	VARCHAR	PK	
Date	Application Date	Date		
StudentNo	Student number	VARCHAR		**FK**
Cnumber	Course Name	VARCHAR		**FK**
CIid	Course Instructor Id	VARCHAR		**FK**
AdminId	Admin number	VARCHAR		**FK**

Figure 4.5: Excuse Application Table

4.5 User Interface Design

The following figures represent the key expected user interfaces of our platform.

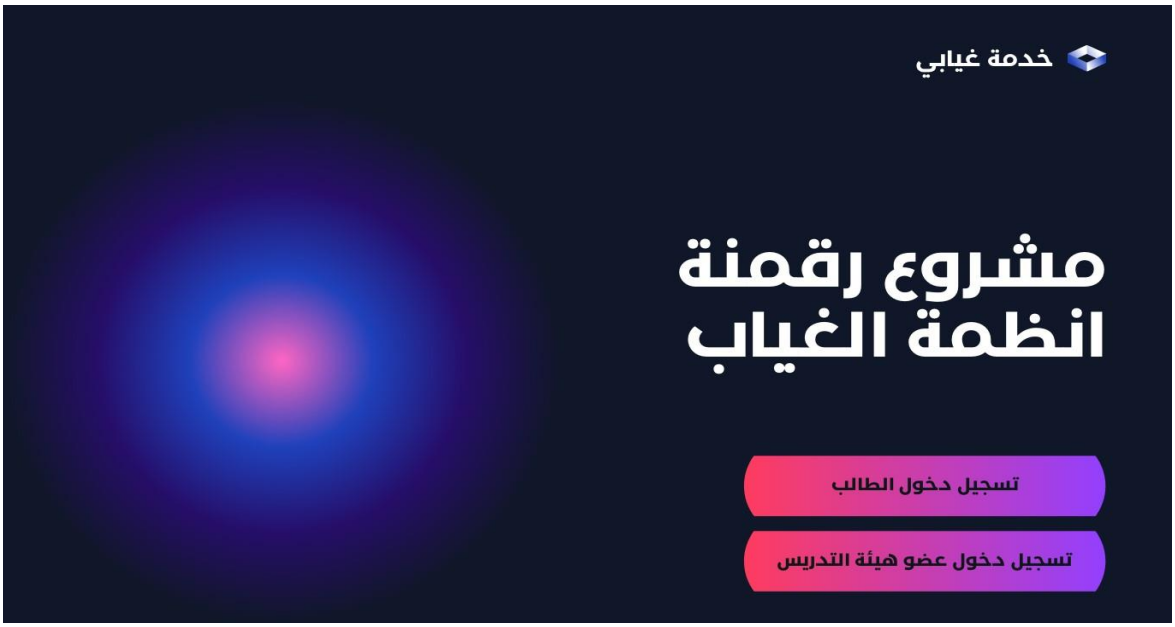

Figure 4.8: Interface of the website

Figure 4.9: Student Log in page Figure **4.10**: Student Sign up page

Figure 4.11: Course Instructore Sign up page

Figure 4.12: Course Instructor Log In page

Figure 4.13: Admin Sign up page

Figure 4.14: Admin Log in page

Figure 4.15: Course schedule page

Figure 4.16: Applications page

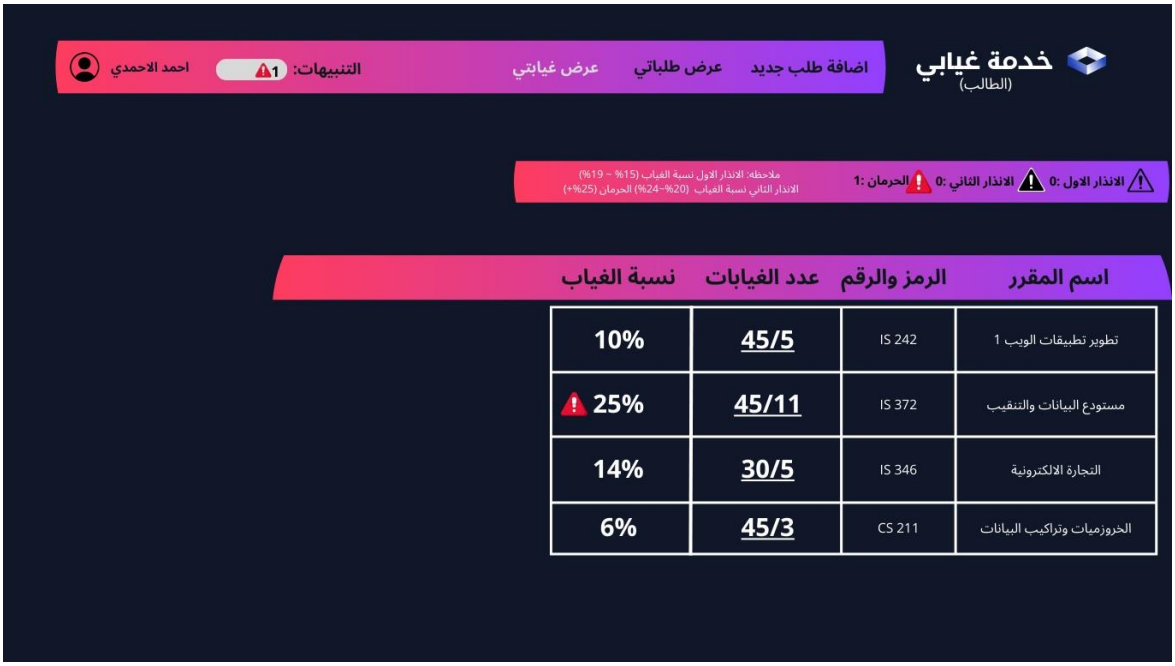

Figure 4.17: absence page

Figure 4.18:Admin Application page

Figure 4.18:Course Instructore Application page

Figure 4.19:Course Instructore Sections page

4.6 Summary

This chapter has represented the system design clearly and concisely in multiple designs, including architectural design, static and dynamic object-oriented designs, and data modeling. Moreover, we have visualized what the user interfaces should look like.

References

[1] "Absence service guide - Qassim University," 2023. [Online]. Available: https://scatalog.qu.edu.sa/827. [Accessed 01 09 2023].

[2] "Absence service guide - Jazan University," 2023. [Online]. Available: https://www.jazanu.edu.sa/ar/colleges/college-engineering/stureq. [Accessed 25 09 2023].

[3] "Call Centeral Berkeley University," 2023. [Online]. Available: https://calcentral.berkeley.edu/. [Accessed 23 09 2023].

Appendix

Questionnaire to collect the opinions of the Student
Link: https://shorturl.at/syEQY