Al-Imam University
Computer Science Department
College of Computer and
Information Science

University Mobile Application for Electronic Services

Course Code: CS290
Assignment: Project
First Semester - Year: 2021

Name: Abdul Karim Al-Luhaidan   440016617
Name: Bassam Al-Ghamdi   440013130
Name: Faisal Alzaydan   440013811
Name: Yasser Al-Ruwaita   440017590
Name: Saleh Al-Obaidan   440016038

Supervisor: Ahmad Misfar Al-Ghamdi

Contents

Project Description:

As a team, we will introduce our services to improve the university by using technology. We will produce an app which provides the university all the services needed. The app will help the professors, students and everybody who works in the university.

The challenge is how we provide the university with the best app because we have huge users with huge services that help them in the app. Maybe it has some trouble with huge users, so we hope to do the best.

Team members:
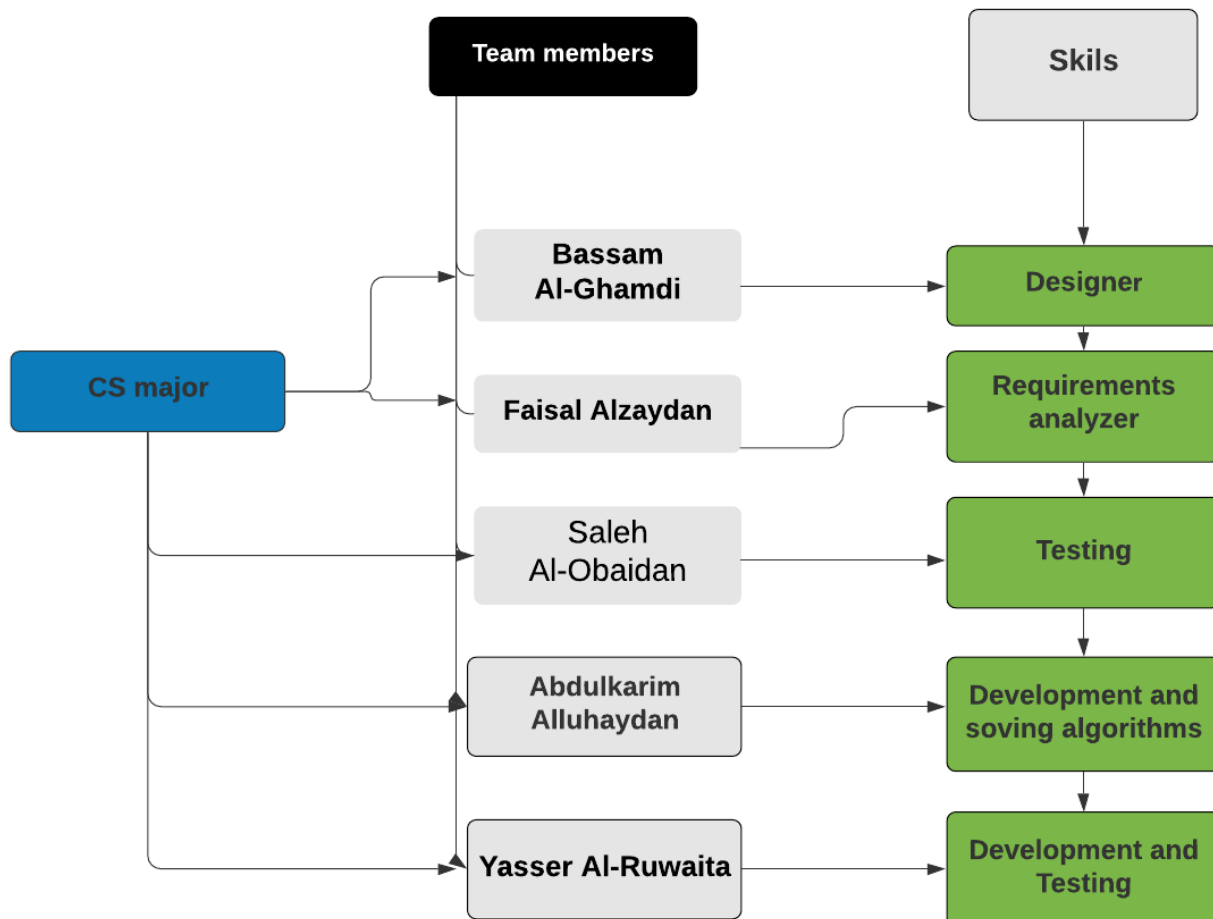
Software Development Lifecycle SDLC of Project is scrum

The Scrum work plan was chosen based on the requirements in the project brochure provided by the University of Hail, where the Scrum work plan is characterized by flexibility and comprehensiveness, which can add modifications to the client's requirements during implementation and is characterized by continuous communication between the team and the client, which helps to show positive results in terms of product quality. and duration of implementation.

Scrum Master	Team members	Project owner
Abdulkarim Alluhaydan	Bassam Al-Ghamdi	Yasser Al-Ruwaita
	Faisal Alzaydan	
	Saleh Al-Obaidan	

Product backlog

Story	estimate	Priority
As an administrator I want to access user's profiles and accounts so I can troubleshoot and edit them as they need	7	1
As a student I want to access my account so I can view my (personal profile, marks, timetable, absences ,GPA ,academic records ,study plan) ,link the timetable to my phone to get notified when having a lecture and being able to edit (add, drop) courses	8	2
As a teacher I want to have access to my account so I can view my (personal profile, timetable, payroll , salary identification certificate) and edit absence	8	2
As an employee I want to access my account so I can view payroll, vacation balance	5	2
As a visitor I want to have a place to follow up my administrative communication	3	3
As an admin I want access to student's academic records and study plans so I can help them continue their studies	6	2

Sprint backlog

Task no.	Task description	Duration (days)	Number of workers
Task 1	Understanding the program and requirements	4	5
Task 2	Identifying requirements and preparing template	3	4
Task 3	Starting to fill the template and detailing the requirements.	4	5
Task 4	Writing BRS (Business requirement system)	3	2
Task 5	Detailing SRS software requirement specification	7	5
Task 6	Gathering everything together to submit the report	2	2
Task 7	Reviewing whole report by all members to submit	3	5

Figure1.1.

Milestones:

18sep

milestone 1

Understanding the program and requirements

22sep

milestone 2

Identifying requirements and preparing template

25sep

milestone 3

Starting to fill the template and detailing the requirements.

29sep

milestone 4

Writing BRS (business requirement system)

2oct

milestone 5

Detailing SRS software requirement specification

9oct

milestone 6

Gathering everything together to submit the report.

12oct

milestone 7

Reviewing whole report by all members to submit

1.Business Requirements Specification

1.1 The purpose of the Project

a) Background

The issue that came up with the idea of making this application was the difficulty for users getting in contact with departments and deanships in the university and guiding them to issue their problems to the right department. By developing an easier way for the users to get in touch with multiple administrations and departments in one application.

b) Goals of the Project

Etimad platform published an announcement from the University of Hail to find a software company that can develop the required project in the required time period and provide all its requirements.

The University of Hail wants to find solutions for communication between students, administration and teachers, and reduce the risks that occur due to a mistake by the administration.

After completing the project, the University of Hail can provide most of the requirements of students, Academic Staff and visitors through the application easily and in less time.

1.2 The Client, the Customer, and other stakeholders.

The Client:

-University of Hail IT department.

The Customer:

1.University of Hail administrators.

2.Students of the university.

3.Academic staff.

4.Visitors.

the Stakeholders

Sponsor   : Hail University members.
Scrum master: Abdulkarim Alluhaydan.
Scrum team: Bassam Al-Ghamdi, Faisal Alzaidan, Yasser Al-Ruwaita, Saleh Al-Obaidan.
Users: students, faculty, staff and visitors.

1.3 Users of the Product

Administration:

Responsible for managing students, professors, adding and deleting subjects. Managing requests from the users.

Student:

User who uses the system to benefit from the services available to him such as displaying the schedule, academic history, etc.

Academic Staff:

Faculty members and academic advising. They have some administrative powers, such as recording student attendance and managing student grades.

Visitors:

Visitors who are interested in applying to the university or even attending a seminar, webinar or a conference.

1.4 The Scope of the Product

The scope of the product includes these main services.

Electronic Inquiries Services:

Profile Review: It should provide the user with all the personal information needed such as his name, age, major, university ID, personal ID, phone number, level and current state.

Review Grades: The student will have access to see his grades.

Academic GPA Review: The student will have access to see his GPA.

View Attendance Record: The student will have access to see his attendance and absence record.

Review Schedule: The student and the faculty members will be able to see the current weekly schedule.

Electronic Reminder: Reminds both the student and the faculty members with dates of the lectures.

Review the Academic Record: The student and the academic advisor will have access to the academic record.

Curriculum Plan: The student and the academic advisor will have access to the curriculum plan.

Attendance Recording: The faculty members will be able to take record of the absence and attendance of students.

View Salary: the employees and the faculty members will have access to view the salary.

Salary Identification Certificate: both the employees and the faculty members will gain access to their personal data such as their name and the date of employment.

Vacation Balance: the employees will have access to view their vacation balance.

Administrative Communications: the visitors will be able to view and track their requests.

University Gate (electronic gate):

General Information Regarding the University: This service will include a thorough information about the university such as a brief of the university history, its graduates, and the benefits of applying to it.

University News: This specific service will tell the user all about the news such as conferences, faculty achievements and general activities.

University Events and Competitions: This service will inform the user with the events and competitions.

1.5 Functional Requirements

How the program works is to provide several services to students, teachers, university employees and visitors in which they can request services and view their information stored in the database.

The maintenance and authorization of the management staff responsible for the users of the system is carried out by the technical team of the system.

Basic requirements for both the electronic services application and the university gate application.

-sign in Sign up.

-Language icon.

- Search box.

- Contact information.

University Gate (electronic gate):

It's for students, faculty, staff and visitors.

- University Twitter account.

-The administration will receive messages from all communicators, students, faculty members, visitors and staff.

-Events and competitions:

-It is published by the university and determines the target users, and the user of the system can register in it if he/she is only the target user.

-Administration Responsible for:

displaying general information about the university and updating it in case of any changes in the future. Publishing university news, ads, events and competitions. setting priorities for those communicating with the university.

Electronic Inquiries Services:

It's for students, faculty, staff and visitors.

Profile Review:

Some information will not be editable such as his/her university ID, personal ID Other information will be subject to change

Review Grades and Academic GPA:
he/she can choose a current or previous semester and it will display the cumulative grade point average, the current grade point average, the grades of the subjects and the number of hours passed.

View Attendance Record:

For each subject, the record of absence will be displayed and what is the maximum amount of absence

Review Schedule: Students and faculty will be able to see their weekly schedules, classrooms and lecture times

Electronic Reminder: The user can turn off this feature and set the time to send the notification, for example, the first user wants to send him an alert half an hour before the lecture, the other user wants to receive an alert an hour before the lecture

Review the Academic Record: Every semester, the academic record is updated by the academic advisor

Curriculum Plan: Students can view the Curriculum Plan only. The academic advisor can modify the Curriculum Plan in the event of any errors or updates.

Attendance Recording: The teacher can record the attendance of students and can view and modify the attendance record for each student.

View Salary: Salary is updated in the event of any changes by H.R.

Salary Identification Certificate: both the employees and the faculty members will gain access to their personal data, but They cannot modify it.

Vacation Balance: Those responsible for the Vacation balance are the human resources, and from this feature, you can apply for sick Vacation or special Vacation.

Administrative Communications: The visitor can enter the system through the ID number and follow up on his requests.

1.6 Non-functional Requirements

Security:

- The permission of the platform data is only accessible by the system data administrator.
- Different levels of login access.
- High data protection procedures used.

Performance:

- Quick to load the page.
- Quick to update the page.
- Quick to refresh.
- Quick to move between pages.

Maintainability:

- The level of difficulty in which maintenance is processed
- All codes must be documented.

Useability:

- The ease of using special functionalities.
- The ease of accomplishing a desired task.

Reliability:

- High protection is used to access specific parts in the system.
- Performing functions under specific conditions.
- The system should shut down completely for any fatal error.

1.7 Business Process Overview

Current Business Process

At any time during or after the application is deployed to support business activities, development/support teams may create user interface elements to test with the customer and user to develop and improve usability of the application.

1. University of Hail officials use the application after each development stage to be sure that it is what they want (requirements validation) and try the UI to see if it is easy to use (ease of use).

2. Determine the budget and time needed to complete the project Allocate tasks to the development team and give enough time to the development team in case of any emergency cases or errors.

2.Software Requirements Specification

2.1 Introduction:

2.1.1 Purpose

Many educational institutions (EI) suffer from a weak online presence, especially reservations and their students pay the price. This system will help EI manage their students' requests and make signing up for students much easier and more reliable. In addition, it eases communications between professors and their students regarding results, missing reports, and much more while ensuring confidentiality and the integrity of the data.

2.1.2 Scope

The university mobile application for electronic services is a mobile application that will mitigate the risks of outdated systems and will facilitate the process of registration easier for students. In addition, the management of the students' records will be manipulated by admins through a small number of steps all done through a mobile application. Furthermore, students will view the university's events, news, and competitions all through news sections in our app.

2.1.3 Definitions and Acronyms

- **UMAFES**: university mobile application for electronic services.
- **Professor**: Any facility teaching member.
- **Student**: Any students registered in a specific university.
- **Admin**: Responsible for the system and having privileged actions.

2.1.4 References

Generating Software Requirements Specification (IEEE- Std. 830- 1998) document with Use Cases and Use case scenarios.

2.1.5 Overview

This SRS document will include the following:

- Description of the **UMAFES** system:
  - Including all factors that affect this product
  - All background information about the product
- Specific requirements

    All types of requirements that are needed to carry out the implementation of the product and help guide the product's design and testing.

2.2 General Description:

2.2.1 Product Perspective

It's a standalone system. It does not depend on any other system or product to function properly.

2.2.2 Product Functions

This subsection of the SRS should provide a summary of the functions that the software will perform.

2.2.3 User Characteristics

The end user of the product will be hail university students and faculty members. Hail university students mostly range between the age of 18 to 24. Practically all of them are using smartphones as in daily base communication. Hail university must use a similar level of proficiency as students already use such a system.

2.2.4 General Constraints

This product must be limited to all extracurriculars occurring related to hail university.

2.3.Functional Requirements or Features

2.3.1 Scenario, Use Case Template, and Associated Requirements

While the IEEE 830-1998 recommends another way of listing out the above segments of the SRS, we believe that following this method can easily trace all requirements without much backtracking and tedious referencing throughout the document. For each use case you will find an ideal scenario and a filled use case template.

Basic requirements for both the electronic services application and the university gate application.

2.3.1.1 Log In.

Use case scenario:

The system allows users to log into their accounts by specifying their usernames and passwords.

The visitor will be allowed to log in using the ID number, phone number and name, the visitor's user interface will have limited options for services intended for visitors only.

Use case template:

ID	B01
Use case Name	Log In
Actors	Admins, Students, Faculty Members
Preconditions	1. Actor has an account stored in the system 2. Actor is not logged into their account 3. Actor specifies desire to log into their account
Postconditions	Actors are logged into their account and can access the system
Normal Flow	1. System requests Login information 2. Actor provides Login information 3. System matches Login information to actor's account 4. System logs actor into their account
Exception Conditions	Actor does not have an account stored in the system Actor provides incorrect Login information
Priority	High

2.3.1.2 Language Icon.

Use case scenario:

The system will allow the user to change the language by pressing the language icon and selecting the language, then the system will change the language.

Use case template:

ID	B02
Use case Name	Language Icon
Actors	Admins, Students, Faculty Members
Preconditions	1. Actor is logged into their account 2. Actor wants to change the language 3.Actor choose the language
Postconditions	The language has changed
Normal Flow	1.The system will display the available languages to the user 2.The system will ask the user to choose the language 3.The system will change the language to the language requested by the user
Exception Conditions	None
Priority	High

2.3.1.3 Search **Box.**

Use case scenario:

The system will allow the user a Search box where the user can enter the desired service and the system will then transfer it to the service directly.

Use case template:

ID	B03
Use case Name	Search Box
Actors	Admins, Students, Faculty Members
Preconditions	1. Actor is logged into their account 2. Actor wants to search for the required service 3.Enter the name of the required service and press the search button
Postconditions	The language has changed
Normal Flow	1.The user interface contains a search box 2.The system will require the user to enter the search information 3.The system will search for the required service 4.The system will transfer the user to the requested service
Exception Conditions	The requested service does not exist
Priority	Medium

University Gate (electronic gate):

2.3.1.4 General Information About the University.

Use case scenario:

The system allows the administrator to add general information about the university or modify some information. Then the system will send a notification to users that there are changes to the university's information.

Use case template:

ID	G01
Use case Name	General Information About the University
Actors	Admins
Preconditions	1. Admin is logged into their account 2. Admin specifies desire to add an information about the university or modify some information information added or information modified
Postconditions	information added or information modified
Normal Flow	1.The system will ask the admin if he wants to edit or add information 2.admin write or modify information 4.The system add, or news modify information
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.5 University News.

Use case scenario:

The system will give the admin permission to add news announcements, faculty achievement, dates of applying and general activities. Then the system will process these requests into the user interface.

Use case template:

ID	G02
Use case Name	University News
Actors	Admins
Preconditions	1.Admin is logged into their account 2.Admin inputs the news and announcements he wants 3.The system will process the requests
Postconditions	System displays the new in UI
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface.
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.6 University Events and Competitions.

Use case scenario:

The system will give the actors permission to add news regarding the events and competitions. Then the system will process these requests into the user interface.

Use case template:

ID	G03
Use case Name	University Events and Competitions
Actors	Admins
Preconditions	1.Admin is logged into their account 2.Admin inputs the news about the events and competitions 3.The system will process the requests
Postconditions	System displays the new in UI
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface.
Exception Conditions	Page title already exists
Priority	High

2.3.1.7 Communicate with the University Administration.

Use case scenario:

The system will allow students, visitors, faculty members and employees to communicate with the university administration. The system will ask the user to enter the address of the message, and then a message

Use case template:

ID	G04
Use case Name	Communicate With the University Administration
Actors	students, visitors, faculty members, employees
Preconditions	1.User logged into his account 2.The user presses the option to communicate with the administration 3.The system will ask the user to enter the address of the message, and then the message 4. Message sent
Postconditions	The administration receives the message from the user
Normal Flow	1.System requests the user to enter the title of the message 2.System requests the user to enter the message and send it 3.The system sent the message to the administration with an indication of the type of sender and the address of the message
Exception Conditions	None
Priority	High

Electronic Inquiries Services:

2.3.1.8 Profile Review.

Use case scenario:

The system will give the actors permission to view their personal data such as name, age, current level, personal ID and the date of enrollment. Then the system will process these requests into the user interface.

Use case template:

ID	S01
Use case Name	Profile Review
Actors	Admins, Students, Faculty staff
Preconditions	1.Actor's log into their account 2.Actors request the data 3.The system will process the requests
Postconditions	The system will display the profile review in the user interface
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.9 Review Grades.

Use case scenario:

Students can view their grade results for the classes that they enrolled in.

Use case template:

ID	S02
Use case Name	Review Grades
Actors	Students
Preconditions	1.Student logs into his account 2.Student make a request to see his grades
Postconditions	The system displays the grades in the user interface
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	High

2.3.1.10 Academic GPA Review.

Use case scenario:

The system will give the student permission to view his GPA.

Use case template:

ID	S03
Use case Name	Academic GPA Review
Actors	Students
Preconditions	1.Student logs into his account 2.Student make a request to see his GPA
Postconditions	The system displays the GPA in the user interface
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.11 View Attendance Record.

Use case scenario:

The system will give the student permission to view his attendance record.

Use case template:

ID	S04
Use case Name	View Attendance Record
Actors	Students
Preconditions	1.Students logs into his account 2.Students make a request to see his attendance record
Postconditions	The system displays the attendance record in the user interface
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	High

2.3.1.12 Review Schedule.

Use case scenario:

The system will give the actors permission to view their schedule.

Use case template:

ID	S05
Use case Name	Review Schedule
Actors	Students, Faculty members
Preconditions	1.Actors logs into their account 2.Actors make a request to view their schedule
Postconditions	The system displays the schedule in the user interface
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	High

2.3.1.13 Electronic Reminder.

Use case scenario:

The system will give the actors permission to turn on the electronic reminder, which will inform them of the lecture date.

Use case template:

ID	S06
Use case Name	Electronic Reminder
Actors	Students, Faculty members
Preconditions	1.Actors logs into their account 2.Actors make a request to turn on their electronic reminder
Postconditions	The system displays the approval of their request
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.14 Review the Academic Record.

Use case scenario:

The system will give the actors permission to view the academic record.

Use case template:

ID	S07
Use case Name	Review the Academic Record
Actors	Students, Academic advisor
Preconditions	1.Actors logs into their account 2.Actors make a request to view their academic record
Postconditions	The system displays the academic record
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.15 Curriculum Plan.

Use case scenario:

The system will give the actors permission to view the curriculum plan.

Use case template:

ID	S08
Use case Name	Curriculum Plan
Actors	Students, Academic advisor
Preconditions	1.Actors logs into their account 2.Actors make a request to view the curriculum plan
Postconditions	The system displays the curriculum plan
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.16 Attendance Recording.

Use case scenario:

The system will give the faculty members permission to record the attendance.

Use case template:

ID	S09
Use case Name	Attendance Recording
Actors	Faculty members
Preconditions	1.Faculty members log into their account 2.Faculty members input the attendance based on the student list
Postconditions	The system will process the inputs and record the attendance
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then execute according to the request
Exception Conditions	Page title already exists
Priority	High

2.3.1.17 View Salary.

Use case scenario:

The system will give the actors permission to view their salary.

Use case template:

ID	S10
Use case Name	View Salary
Actors	Faculty members, Admins
Preconditions	1.Actors logs into their account 2.Actors make a request to view their salary
Postconditions	The system displays the salary
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.18 Salary Identification Certificate.

Use case scenario:

The system will give the actors a certificate regarding their salary identity.

Use case template:

ID	S11
Use case Name	Salary Identification Certificate
Actors	Faculty members, Admins
Preconditions	1.Actors logs into their account 2.Actors make a request to get a salary identification certificate
Postconditions	The system displays the certificate
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.19 Vacation Balance.

Use case scenario:

The system will give the admins permission to view their vacation balance.

Use case template:

ID	S12
Use case Name	Vacation Balance
Actors	Admins
Preconditions	1.Admins logs into their account 2.Admins make a request to view their vacation balance
Postconditions	The system displays the balance
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	Medium

2.3.1.20 Administrative Communications.

Use case scenario:

The system will give the visitors permission to ask questions regarding the university and track their previous requests.

Use case template:

ID	S13
Use case Name	Administrative Communications
Actors	Visitors
Preconditions	1.Visitors can make a request regarding the university 2.Visitors can track the request.
Postconditions	The system displays the news regarding the request
Normal Flow	1.The system will take the requests 2.The system will process the requests 3.The system will then display it by the user interface
Exception Conditions	Page title already exists
Priority	High

2.3.2 Use Case Diagrams:

2.3.2.1 University Gate.
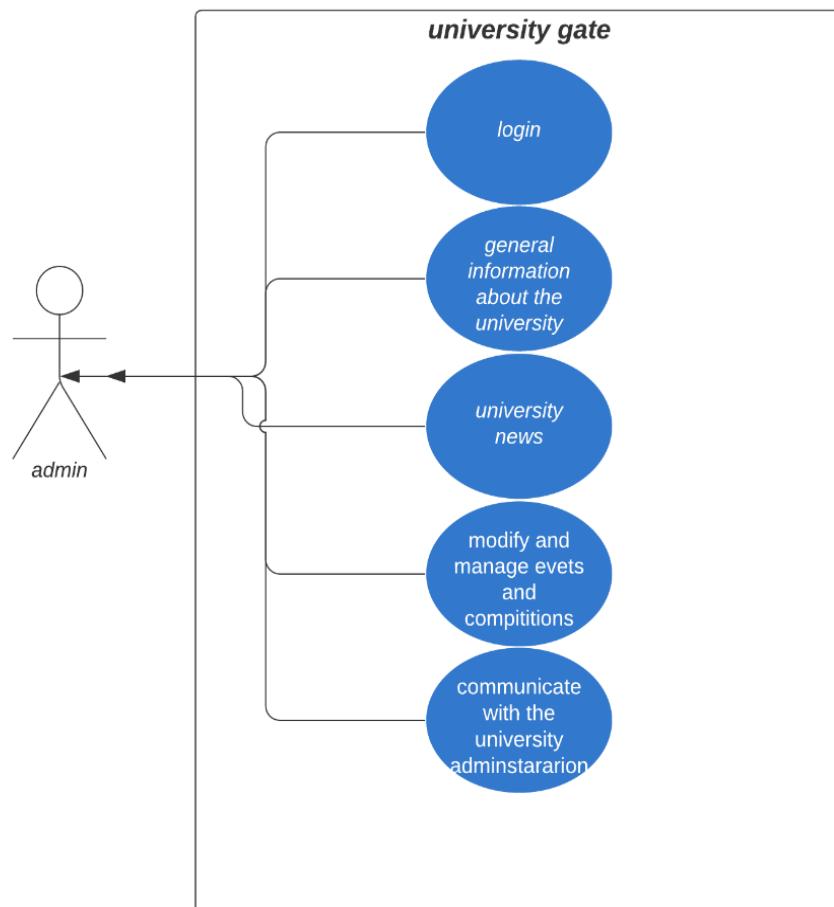
Figure 1.2: admin-university gate use-case diagram.

university gate

- login
- general information about the university
- university news
- university events and compititions
- communicate with the university adminstararion
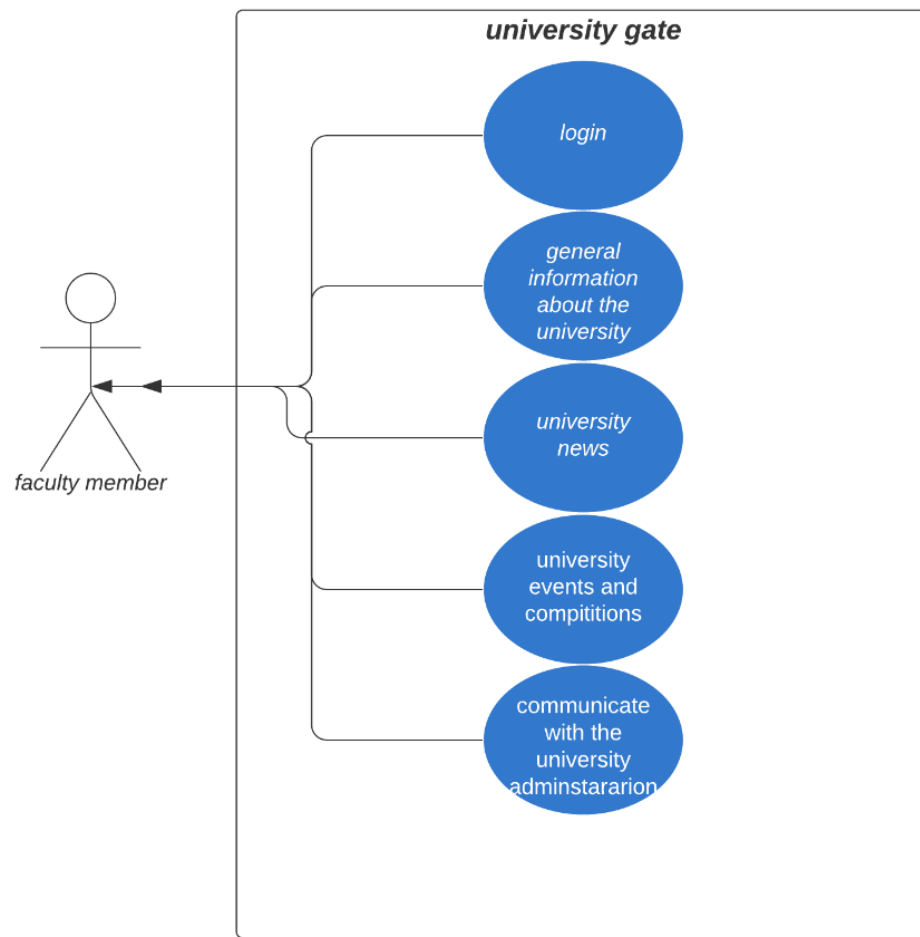
faculty member

Figure 1.3: faculty member-university gate use-case diagram.
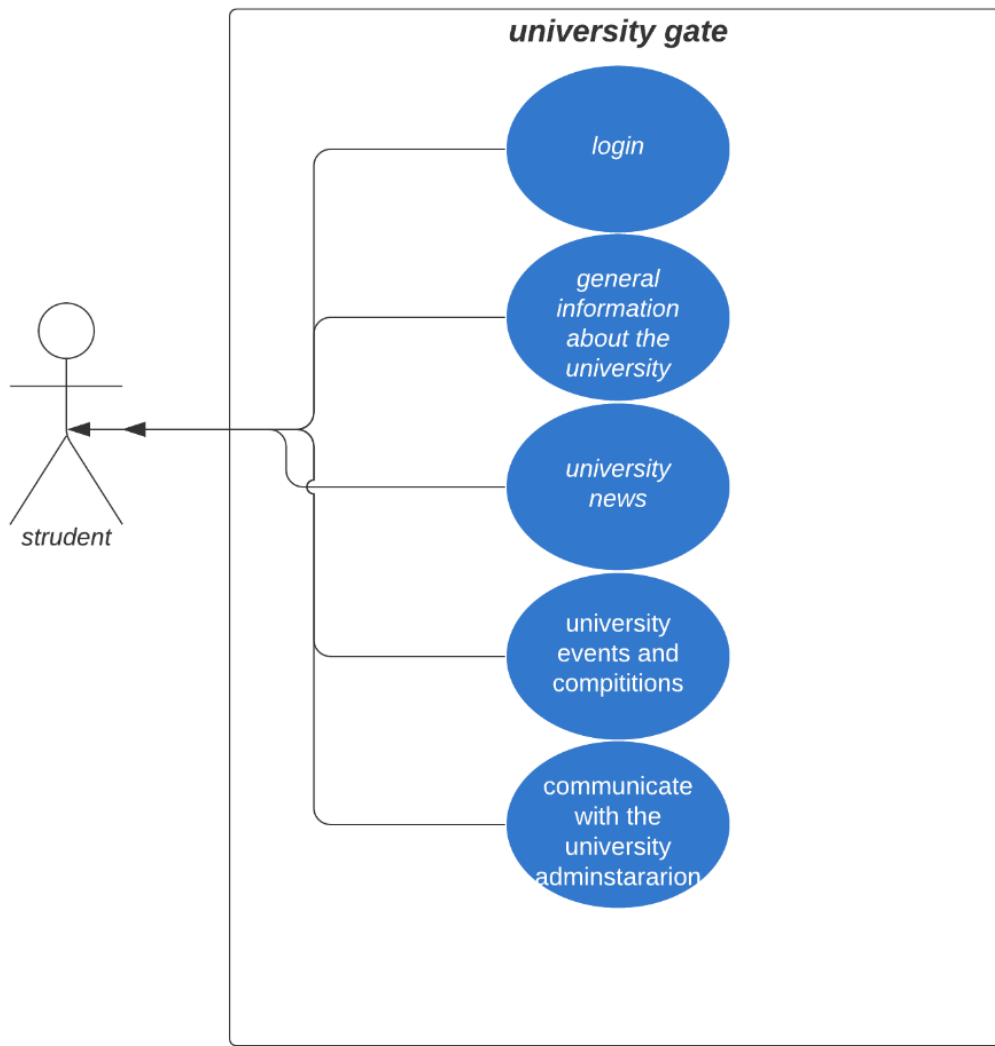
Figure 1.4: visitor-university gate use-case diagram.

2.3.2.2 Electronic Inquiries Services
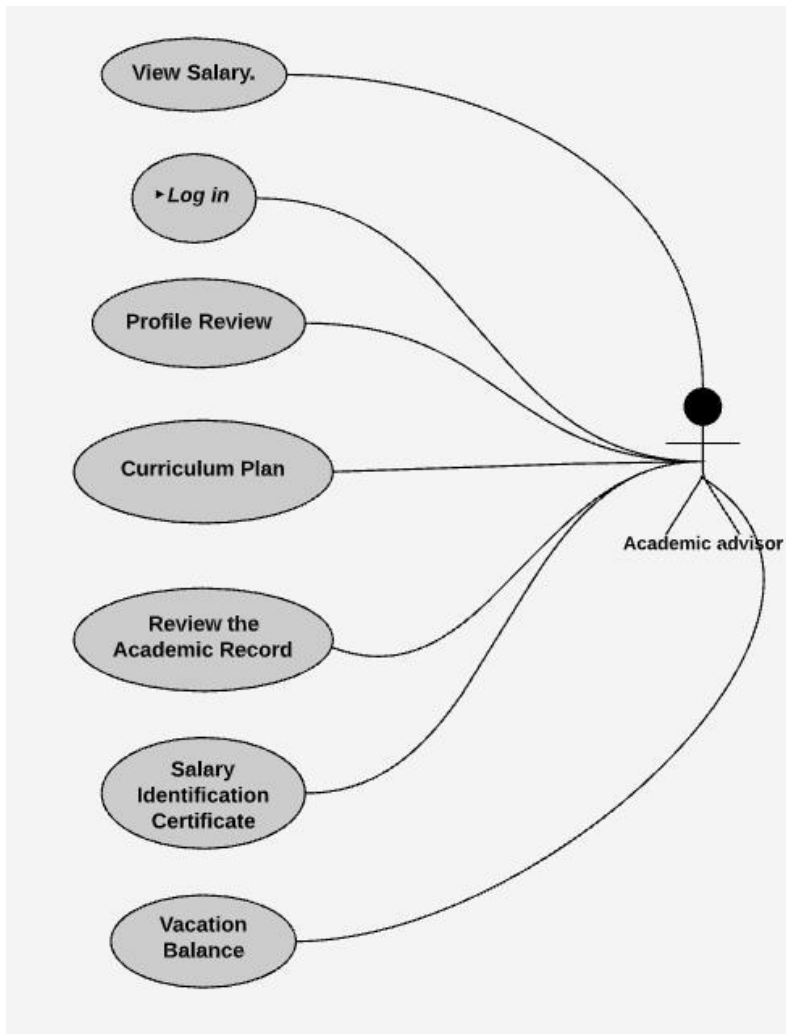
Figure 1.6: academic advisor-Electronic Inquiries Services use-case diagram.
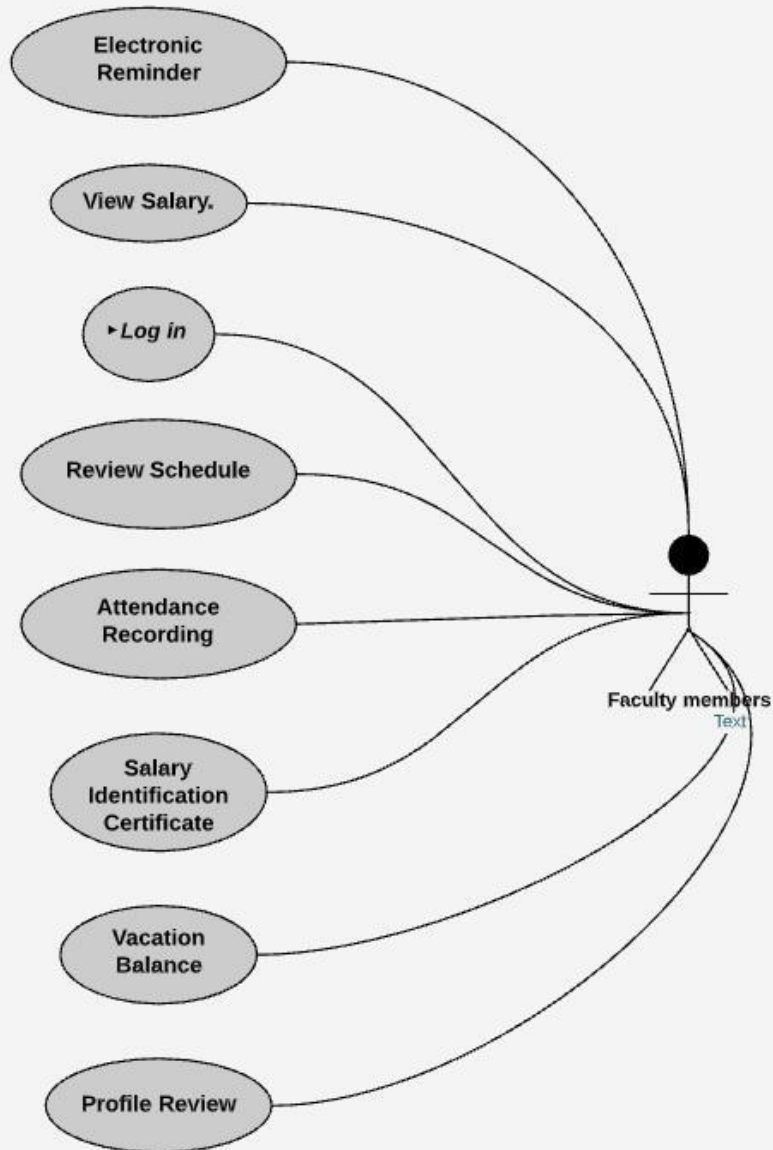
Figure 1.7: faculty member- Electronic Inquiries Services use-case diagram.
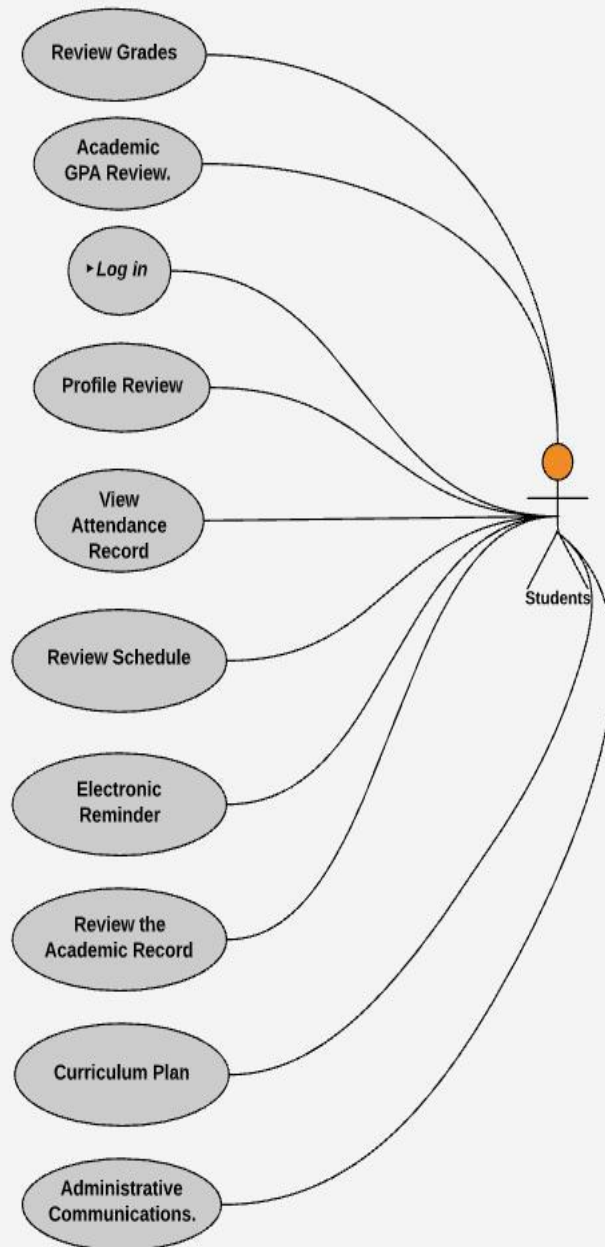
Figure 1.8: student- Electronic Inquiries Services use-case diagram.
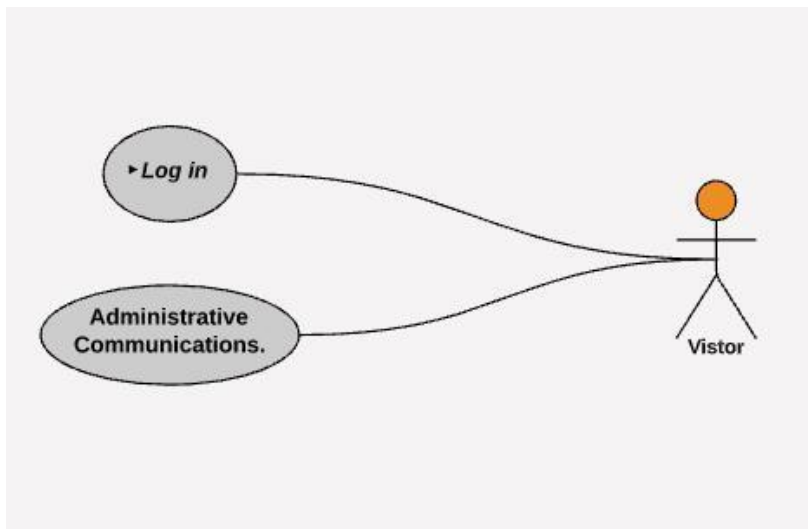
Figure 1.9: visitor- Electronic Inquiries Services use-case diagram.

2.3.3 Non-functional Requirements:

2.3.3.1 Performance:

- Quick to load the page.
- Quick to update the page.
- Quick to refresh.
- Quick to move between pages.

2.3.3.2 Reliability:

- High protection is used to access specific parts in the system.
- Performing functions under specific conditions.
- The system should shut down completely for any fatal error.

2.3.3.3 Useability:

- The ease of using special functionalities.
- The ease of accomplishing a desired task.

2.3.3.4 Security:

- The permission of the platform data is only accessible by the system data administrator.
- Different levels of login access.
- High data protection procedures used.

2.3.3.5 Maintainability:

- The level of difficulty in which maintenance is processed
- All codes must be documented.

2.3.4 Logical Database Requirements:

1.The system shall give access for information files stored on multiple hard drives from a single user interface.
2.The system must store all user account information. All the data shall be stored in text-based files.
3.Each user account information shall be stored in one file containing the login ID, username, email and password.

2.3.5 Other Requirements:

Clarification of the 55 services provided by the university

1-scholarship request: This service enables faculty members to request a scholarship.

2-promoting to associate professor: This service enables faculty members to request a promotion to associate professor.

3- promoting a professor request: This service enables faculty members to request a promotion to a professor.

4- request to designate teaching assistant as a lecturer: This service enables faculty members to be a lecturer

5-request to designate teaching assistant as an attributed professor: This service enables faculty members to be an attributed professor

6- request to designate teaching: assistant as a non-attributed professor This service enables faculty members to be a non-attributed professor

7- request for participating in a conspiracy or seminar: This service enables faculty members to participate in a conspiracy or seminar

8- request for participating in an external training course: This service enables faculty members to participate in an external course

9- extending scholarship request: This service enables faculty members to request time extension for their scholarship

10- scientific trip request: This service enables faculty members to request a scientific trip

11-Request to attend a seminar or conference: This service enables faculty members to apply to attend a seminar or conference inside the Kingdom.

12-Request to change the language institute: This service enables faculty members and university employees on scholarships abroad to request a change of the language institute.

13-Request to modify the end of the mission: This service enables faculty members and university employees on scholarships abroad to apply for a scholarship promotion request to study at a higher stage than the current stage.
1- Submitting a request for a mission promotion through the Ambassador program.
2- The approval of the attaché and the university on the request.
3- Follow up the application on the university's electronic portal.

14-Request to continue studying the language: This service enables faculty members and university employees to apply to continue studying the language.

15-Request general requests: This service enables faculty members and university employees to request any general requests.

16-Request training courses: This service enables faculty members and university employees on scholarships abroad to apply for a training course according to the university's rules and regulations.
1- Submit an application on the Safir system.
2- The approval of the attaché and the university on the request.

17-University change request: This service enables faculty members and university employees on scholarships abroad to apply for changing from one university to another according to the university's rules and regulations.
1- Submitting a request to change the university or the scholarship country through the Safir program.
2- Approval of the attaché on the request.

18-Request to change specialization for a scholarship student: This service enables faculty members and university employees on scholarships abroad to apply for a change of major.

1- Submitting a request for a change of specialization through the Ambassador program.

2- The approval of the attaché and the university on the request.

3- Follow up the application on the university's electronic portal.

19-Request to transfer from a country: This service enables faculty members and university employees on scholarships abroad to apply to change the scholarship country and move to another country according to the university's rules and regulations.

Measures:

1- Submit an application on the Safir system.

2- The approval of the attaché and the university on the transfer request.

3- Follow up the application on the university's electronic portal.

20-Request to postpone a mission: This service enables faculty members and university employees on scholarships abroad to submit a request to postpone a scholarship according to the university's rules and regulations.

1- Submitting a request to postpone the mission through the Ambassador program.

2- The approval of the attaché and the university on the request.

3- Follow up the application on the university's electronic portal.

21-Request to end a mission: This service enables faculty members and university employees on scholarships abroad to apply for a scholarship in accordance with the university's rules and regulations.

1- Submitting a request to terminate a scholarship through the Safir Program.

2- Follow up the application on the university's electronic portal.

22-Student Identification: Certificate: This service will allow the student to obtain a certificate regarding the student's information. His full name, student ID and age.

23-Request an Online Class: This service will allow the student to send a request to take classes online.

24-Upgrading a Scholarship: This service will allow the scholar to upgrade his scholarship if the conditions allow him to do so.

25-Cancel Request: This service gives the user permission to cancel a request.

26-Scholarship Resumption: This service gives the user permission to resume his scholarship.

27-Employee Scholarship Request: This service will allow the employee to ask for a scholarship.

28-Migrate Allowance: This service will allow the user to send a request for a one-month salary.

29-A One-Way Ticket Allowance: This service will allow the scholar to request a one-way ticket.

30-Round Trip Allowance: This service will allow the scholar to request a round trip.

31-Return Ticket Allowance: This service will allow the scolar to request a return ticket.

32-Printing Allowance: This service will allow the user to send a request to print out a document.

33,34,35- Books and Sources Allowance: This service will allow the user to send a request for a one-month salary three times.

36-Request for Excellence Award for Scientific Publication: This service allows faculty members to request a reward for excellence for scientific publication.

37-Scientific Communication Request: This service is available for faculty members to go on a scientific mission, teach outside the Kingdom, or conduct scientific research at another university in summer.

38-Scientific mission report: This service allows the teaching member, after completing the scientific mission, to assign the mission report.

39-Request to enable starting work: This service allows the new employee to request empowerment to start work.

40-Scientific Publishing Request: This service allows the author of a scientific research to publish it.

41-Sabbatical application: This service allows a faculty member to request a sabbatical vacation, provided that the application is submitted six months before the vacation.

42-Registering a topic and appointing a supervisor: This service allows faculty members to register a new topic and appoint a supervisor.

43-Formation of a discussion committee: This service allows faculty members to form a discussion committee on scientific research and assign important points.

44-Obtaining the approval of the Permanent Committee for the Ethics of Scientific Research at the University of Hail: This service allows the researcher to submit research to the university and the university will issue approval if the research meets the requirements of ethics.

45-Research groups: This service allows the university to see all research groups.

46-Apply to (badee'): this service is applying your research to the university for approval, and it allows all people in the university, and this has terms:
-No more than two years have passed since the appointment of a faculty member at the rank of assistant professor at the university.
-The research should not be extracted from the PhD thesis.
-The research should not have been completed previously, in whole or in part.
-The research proposal must be submitted in detail according to the forms designated for this, which can be obtained from the website of the Deanship of Scientific Research.
- That the research proposal be subject to scientific arbitration, followed by signing an agreement to finance a research project with the Deanship of Scientific Research at the university.

47- Check quotation ratio: this service is to check if the research is quoted with percentage of quotation ratio, it is for a faculty member, and he can't use it more than 3 times in a year.

48- Institutional Finance Initiative: this service is allowed to have Institutional Finance, it is for students, and the submitted proposal should be among the priorities.

49- Research proposal arbitration form: this service is to have the research proposal arbitration form for students or who want a research proposal arbitration form.

50- Request to close a research project: this service for researchers to close their research project.

51- Research project extension request: this service for researchers to request an extension of time.

52- Request to transfer a research project: this service for researchers to request to transfer the research.

53- Request to cancel a research project: this service for researchers to close their research project.

54- Master's and Diploma Programs: this service for visitors or those interested with master's and diploma Programs, and it shows the program.

55-University Council system: this is a system for University Council members, and it provides all services they need.

55- Legal management system: this is a system to the Legal management members, and it provides all services they need. 61-Polarization system: this system for university administration and it provides all services they need. 62-Teaching assistants and lecturer jobs: this service is for teaching assistants and lecturers to apply for a job. 63- Assistant Professor - Associate - Professor Jobs: this service is for Assistant Professor - Associate - Professor to apply for a job

2.4. Change Management Process

The SRS will change when project scope or requirements are needed to change, and before the change the team will discuss and make the decision to change.

3.Phase 2

3.1 Introduction:

The aim of this document is to provide a detailed description of "University Mobile Application for Electronic Services" . It will show the purpose and full declaration for the system's development. It will also explain system's limitations, interface, and interactions. This document is specifically designed to be presented to a customer for acceptance and to serve as a guide for designing the first version.

Product backlog

Story	estimate	priority
As an administrator I want to access user'sprofiles and accounts so I can troubleshoot and edit them as they need	3	1
As a student I want to access my account so I can view my (personal profile ,marks ,timetable ,absences ,GPA ,academic records ,study plan) ,link the timetable to my phone to get notified when having a lecture and being able to edit (add,drop) courses	5	2
As a teacher I want to have access to my account so I can view my (personal profile,timetable,payroll, salary identification certificate) and edit absence	2	2
As an employee I want to access my account so I can view payroll, vacation balance	2	2
As a visitor i want to have a place to follow up my administrative communication	4	3
As an admin I want access to student's academic records and study plans so I can help them continue their studies	3	2

Sprint backlog

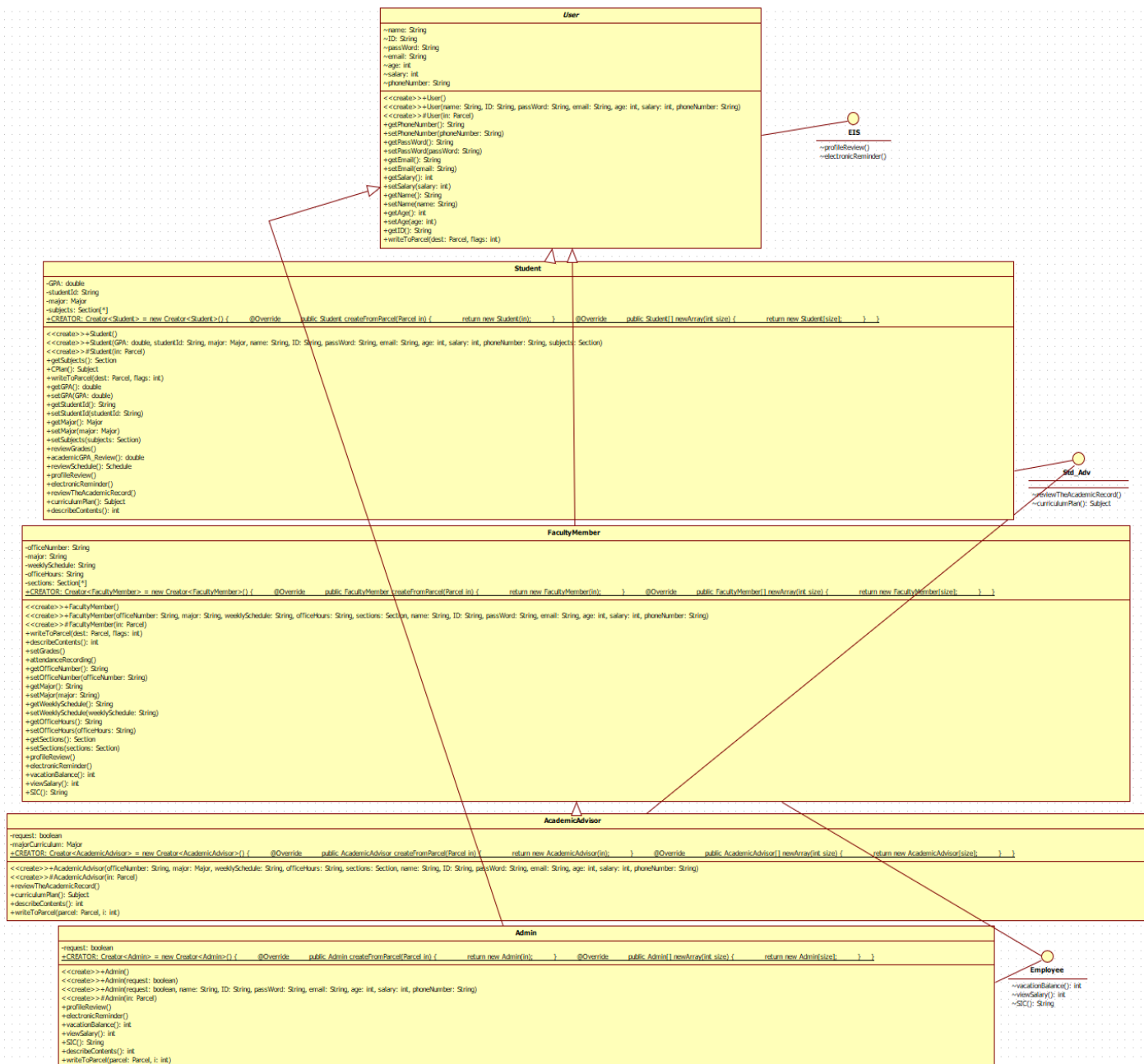| Task no.	Task description	Duration (days)	Number of workers
Task 1	Understanding the design requirements	1	5
Task 2	Designing the system architecture (Class Diagram)	4	3
Task 3	Implementing the flowchart	2	1
Task 4	Choosing the suitable MVC pattern	3	2
Task 5	Making the use case diagrams	1	2
Task 6	Designing the user interface prototype	3	2
Task 7	Reviewing whole report by all members to submit	1	5

3.2 Class Diagram:

User

~name: String
~ID: String
~passWord: String
~email: String
~age: int
~salary: int
~phoneNumber: String

<<create>>+User()
<<create>>+User(name: String, ID: String, passWord: String, email: String, age: int, salary: int, phoneNumber: String)
<<create>>#User(in: Parcel)
+getPhoneNumber(): String
+setPhoneNumber(phoneNumber: String)
+getPassWord(): String
+setPassWord(passWord: String)
+getEmail(): String
+setEmail(email: String)
+getSalary(): int
+setSalary(salary: int)
+getName(): String
+setName(name: String)
+getAge(): int
+setAge(age: int)
+getID(): String
+writeToParcel(dest: Parcel, flags: int)

EIS

~profileReview()
~electronicReminder()

Student

-GPA: double
-studentId: String
-major: Major
-subjects: Section[*]
+CREATOR: Creator<Student> = new Creator<Student>() {   @Override   public Student createFromParcel(Parcel in) {   return new Student(in);   }   @Override   public Student[] newArray(int size) {   return new Student[size];   }   }

<<create>>+Student()
<<create>>+Student(GPA: double, studentId: String, major: Major, name: String, ID: String, passWord: String, email: String, age: int, salary: int, phoneNumber: String, subjects: Section)
<<create>>#Student(in: Parcel)
+getSubjects(): Section
+CPlan(): Subject
+writeToParcel(dest: Parcel, flags: int)
+getGPA(): double
+setGPA(GPA: double)
+getStudentId(): String
+setStudentId(studentId: String)
+getMajor(): Major
+setMajor(major: Major)
+setSubjects(subjects: Section)
+reviewGrades()
+academicGPA_Review(): double
+reviewSchedule(): Schedule
+profileReview()
+electronicReminder()
+reviewTheAcademicRecord()
+curriculumPlan(): Subject
+describeContents(): int

Std_Adv

~reviewTheAcademicRecord()
~curriculumPlan(): Subject

FacultyMember

-officeNumber: String
-major: String
-weeklySchedule: String
-officeHours: String
-sections: Section[*]
+CREATOR: Creator<FacultyMember> = new Creator<FacultyMember>() {   @Override   public FacultyMember createFromParcel(Parcel in) {   return new FacultyMember(in);   }   @Override   public FacultyMember[] newArray(int size) {   return new FacultyMember[size];   }   }

<<create>>+FacultyMember()
<<create>>+FacultyMember(officeNumber: String, major: String, weeklySchedule: String, officeHours: String, sections: Section, name: String, ID: String, passWord: String, email: String, age: int, salary: int, phoneNumber: String)
<<create>>#FacultyMember(in: Parcel)
+writeToParcel(dest: Parcel, flags: int)
+describeContents(): int
+setGrades()
+attendanceRecording()
+getOfficeNumber(): String
+setOfficeNumber(officeNumber: String)
+getMajor(): String
+setMajor(major: String)
+getWeeklySchedule(): String
+setWeeklySchedule(weeklySchedule: String)
+getOfficeHours(): String
+setOfficeHours(officeHours: String)
+getSections(): Section
+setSections(sections: Section)
+profileReview()
+electronicReminder()
+vacationBalance(): int
+viewSalary(): int
+SIC(): String

AcademicAdvisor

-request: boolean
-majorCurriculum: Major
+CREATOR: Creator<AcademicAdvisor> = new Creator<AcademicAdvisor>() {   @Override   public AcademicAdvisor createFromParcel(Parcel in) {   return new AcademicAdvisor(in);   }   @Override   public AcademicAdvisor[] newArray(int size) {   return new AcademicAdvisor[size];   }   }

<<create>>+AcademicAdvisor(officeNumber: String, major: Major, weeklySchedule: String, officeHours: String, sections: Section, name: String, ID: String, passWord: String, email: String, age: int, salary: int, phoneNumber: String)
<<create>>#AcademicAdvisor(in: Parcel)
+reviewTheAcademicRecord()
+curriculumPlan(): Subject
+describeContents(): int
+writeToParcel(parcel: Parcel, i: int)

Admin

-request: boolean
+CREATOR: Creator<Admin> = new Creator<Admin>() {   @Override   public Admin createFromParcel(Parcel in) {   return new Admin(in);   }   @Override   public Admin[] newArray(int size) {   return new Admin[size];   }   }

<<create>>+Admin()
<<create>>+Admin(request: boolean)
<<create>>+Admin(request: boolean, name: String, ID: String, passWord: String, email: String, age: int, salary: int, phoneNumber: String)
<<create>>#Admin(in: Parcel)
+profileReview()
+electronicReminder()
+vacationBalance(): int
+viewSalary(): int
+SIC(): String
+describeContents(): int
+writeToParcel(parcel: Parcel, i: int)

Employee

~vacationBalance(): int
~viewSalary(): int
~SIC(): String

Figure 2.1: Class Diagram for the Electronic Inquiries Services.

3.3 Class Diagram Continuous:

Visitor

-name: String
-phoneNumber: String
-email: String
-administartion: Admin

<<create>>+Visitor()
<<create>>+Visitor(name: String, phoneNumber: String, email: String)
+getName(): String
+setName(name: String)
+getPhoneNumber(): String
+setPhoneNumber(phoneNumber: String)
+getEmail(): String
+setEmail(email: String)
+getAdministartion(): Admin
+setAdministartion(administartion: Admin)
+administrativeCommunications()

College

-dean: FacultyMember
-majors: Major[*]
-contactEmail: String

<<create>>+College()
<<create>>+College(dean: FacultyMember, majors: Major, contactEmail: String)
+contactCollege()

Schedule

-schedule: Subject[*]
-dates: Date[*]

+showSchedule(): String

Date

-day: int
-month: int
-year: int
+CREATOR: Creator<Date> = new Creator<Date>() {   @Override   public Date createFromParcel(Parcel in) {   return new Date(in);   }   @Override   public Date[] newArray(int size) {   return new Date[size];   } }

<<create>>+Date(day: int, month: int, year: int)
<<create>>#Date(in: Parcel)
+toString(): String
+describeContents(): int
+writeToParcel(parcel: Parcel, i: int)

Section

+input: Scanner = new Scanner(System.in)
-teacher: FacultyMember
-subject: Subject
-students: String[*]
+time: int[*]
+day: int[*]
-studentsGrades: int[*]
-studentsattendace: int[*]
-studentsCounter: int = 0
-sectionNumber: String
-classRoom: String
-prohibtionLimit: int
-finalGrade: String[*]
+CREATOR: Creator<Section> = new Creator<Section>() {   @Override   public Section createFromParcel(Parcel in) {   return new Section(in);   }   @Override   public Section[] newArray(int size) {   return new Section[size];   } }

<<create>>+Section()
<<create>>+Section(subject: Subject, sectionNumber: String, classRoom: String)
<<create>>#Section(in: Parcel)
+writeToParcel(dest: Parcel, flags: int)
+describeContents(): int
+addStudents(id: String)
+addTeacher(teacher: FacultyMember)
+calculateFinalGrades()
+viewGrade(id: String): String
+setGrades()
+takeAttendance()
+getTeacher(): FacultyMember
+setTeacher(teacher: FacultyMember)
+getSubject(): Subject
+getTime(): int
+setTime(time: int)

Subject

-name: String
-subjectCode: String
-creditHours: int
-date: Date
+CREATOR: Creator<Subject> = new Creator<Subject>() {   @Override   public Subject createFromParcel(Parcel in) {   return new Subject(in);   }   @Override   public Subject[] newArray(int size) {   return new Subject[size];   } }

<<create>>+Subject()
<<create>>+Subject(name: String, subjectCode: String, creditHours: int)
<<create>>#Subject(in: Parcel)
+getDate(): Date
+setDate(date: Date)
+getName(): String
+setName(name: String)
+getSubjectCode(): String
+setSubjectCode(subjectCode: String)
+getCreditHours(): int
+setCreditHours(creditHours: int)
+describeContents(): int
+writeToParcel(parcel: Parcel, i: int)

Major

~name: String
-curriculum: Subject[*]
-academicStaff: FacultyMember[*]
-chairman: FacultyMember
+CREATOR: Creator<Major> = new Creator<Major>() {   @Override   public Major createFromParcel(Parcel in) {   return new Major(in);   }   @Override   public Major[] newArray(int size) {   return new Major[size];   } }

<<create>>+Major()
<<create>>+Major(name: String, curriculum: Subject, academicStaff: FacultyMember, chairman: FacultyMember)
<<create>>#Major(in: Parcel)
+writeToParcel(dest: Parcel, flags: int)
+describeContents(): int
+getMName(): String
+setName(name: String)
+getCurriculum(): Subject
+setCurriculum(curriculum: Subject)
+getAcademicStaff(): FacultyMember
+setAcademicStaff(academicStaff: FacultyMember)
+getChairman(): FacultyMember
+setChairman(chairman: FacultyMember)

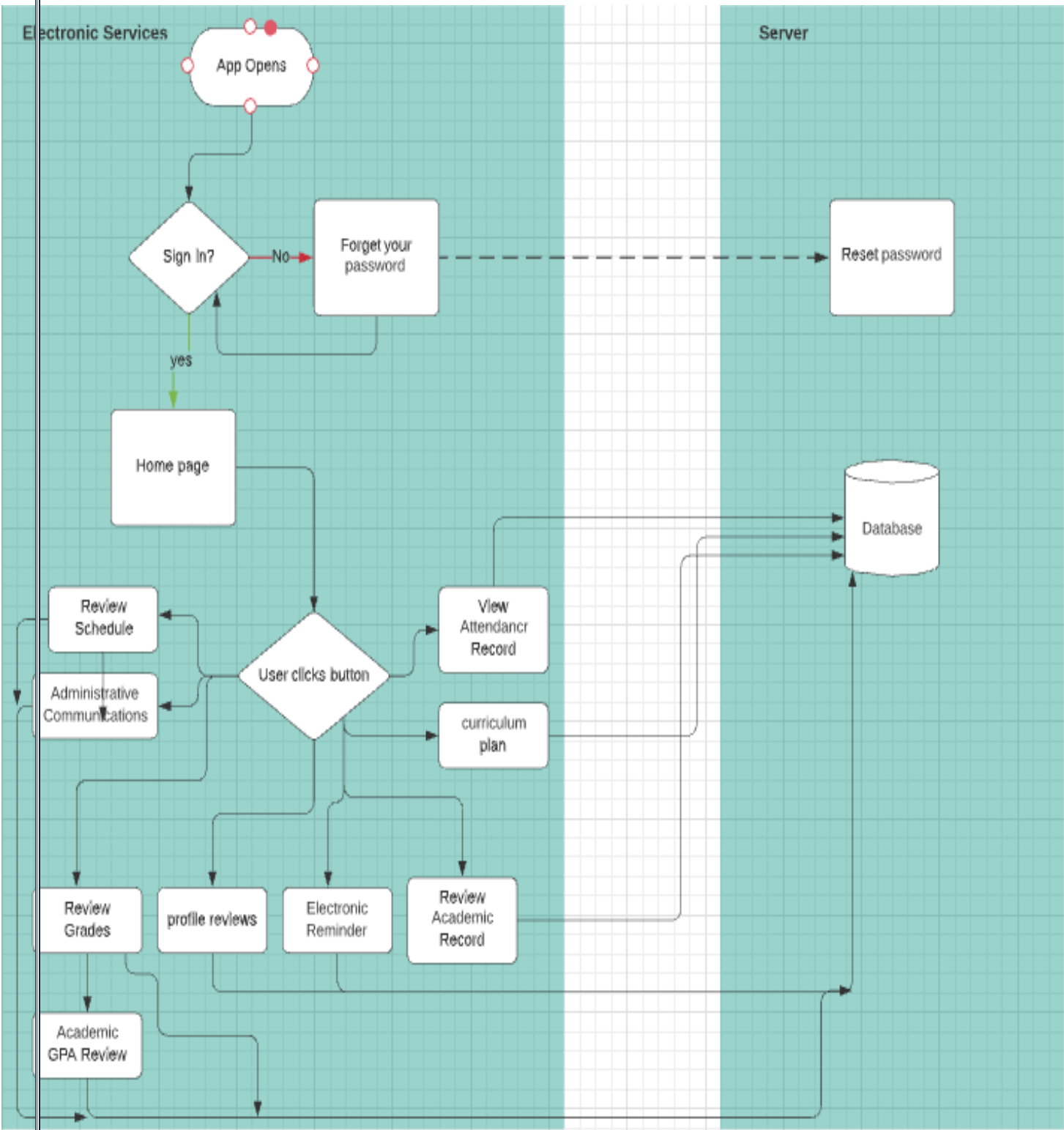Figure 2.2: Class Diagram for the Electronic Inquiries Services.

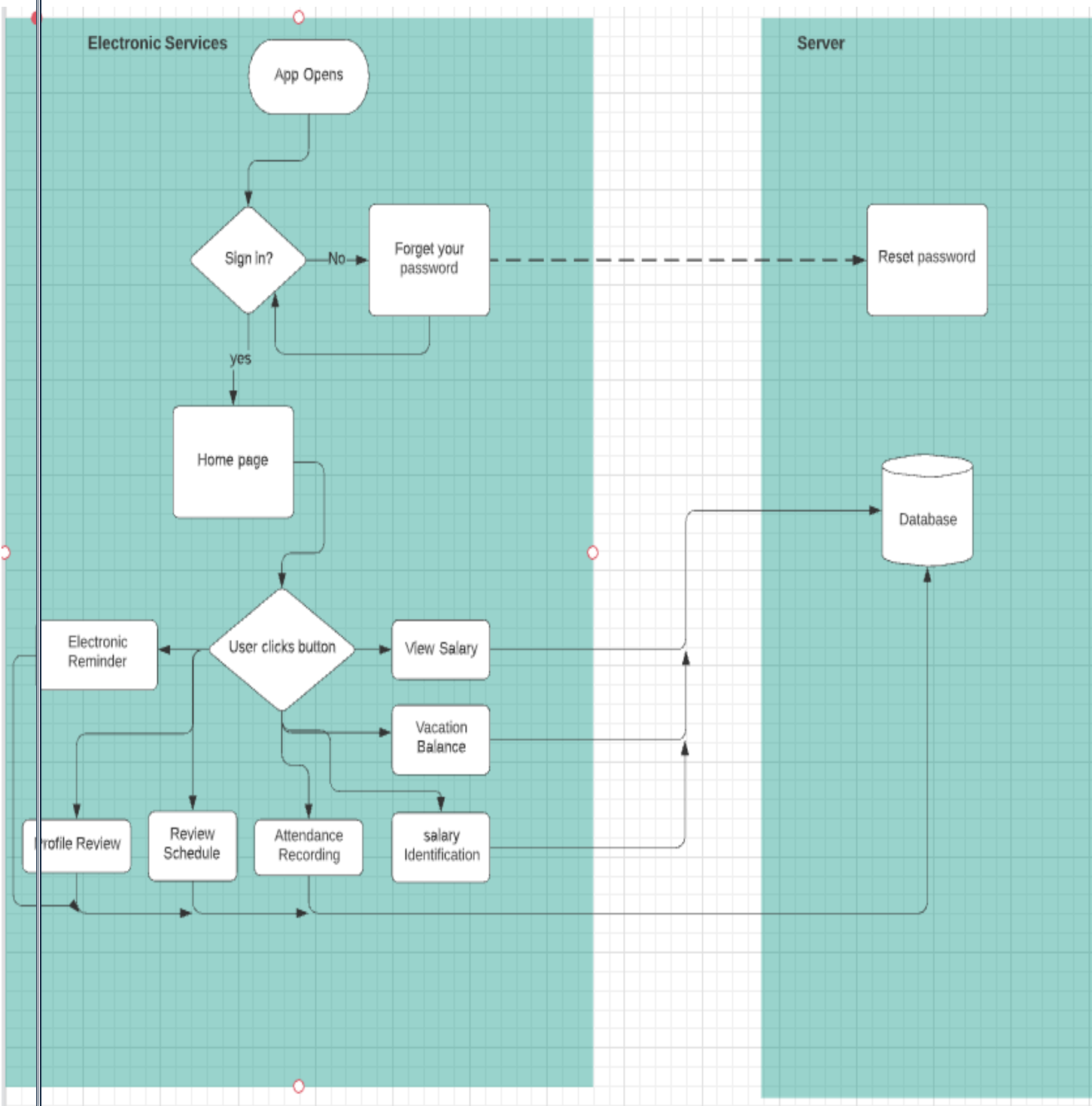Figure 2.3:Flow Chart for student page.
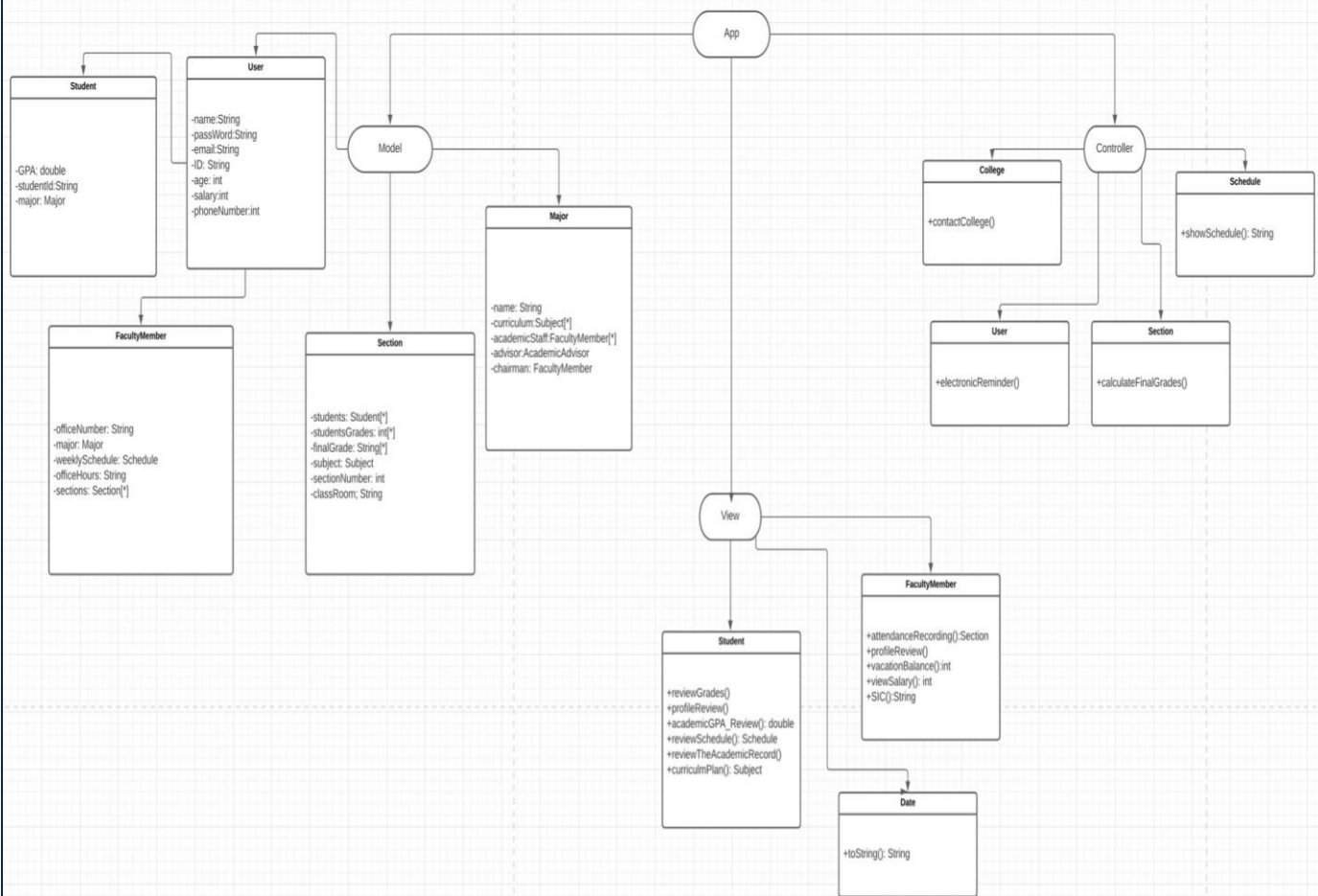
Figure 2.4: Flow Chart for Faculty members page.

Figure 2.5 : MVC Pattern for the Electronic Inquiries Services Design.

- We choose the MVC architectural design because it suits our application requirements. Since we are asked to create a functional phone app rather than a website for instance it's more convenient.
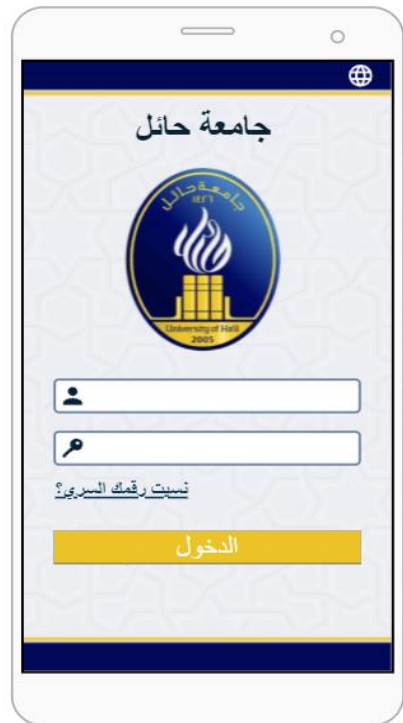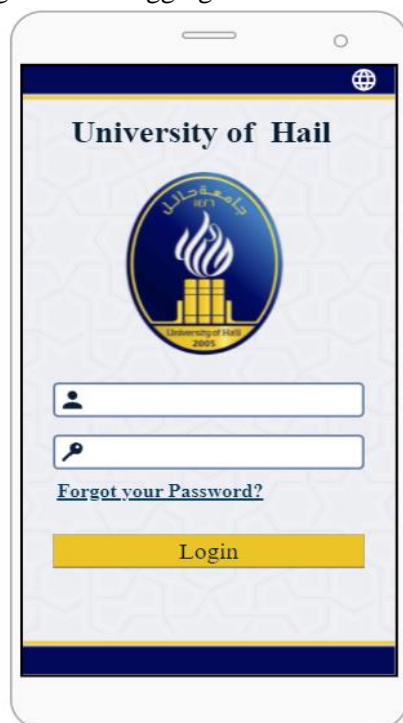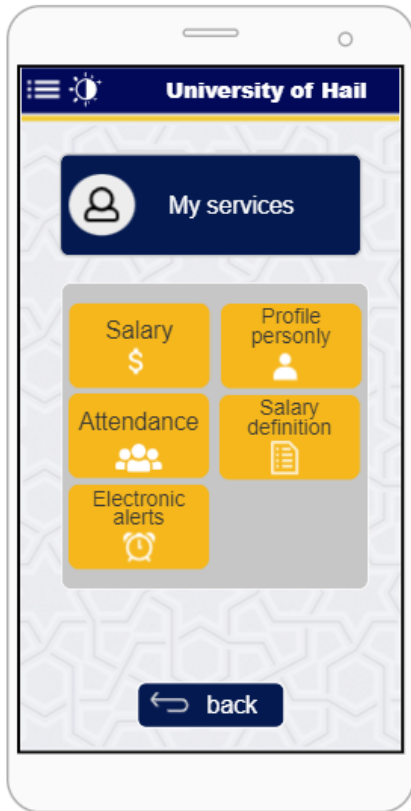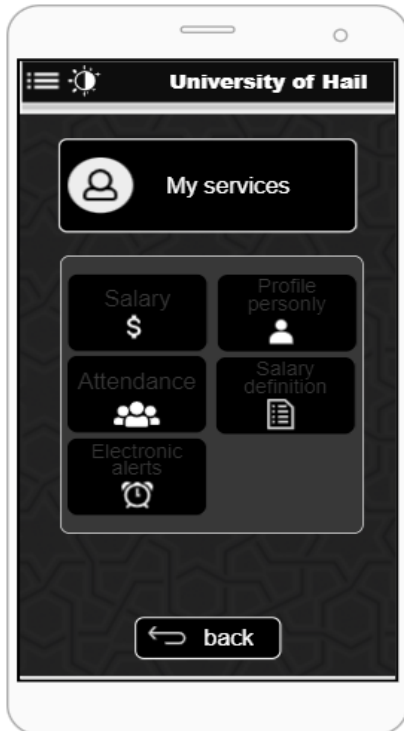
Figure 2.6 : Logging-in interface.

Figure2.7 : Change Language interface.

List of Services For faculty members

Figure 2.8 : List of Services

Dark mood

Figure 2.9 : Dark mood
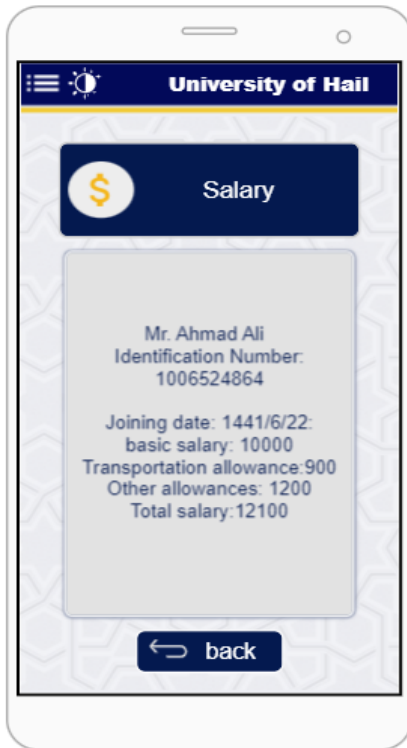
Salary information

Figure 2.10: Salary information.

.
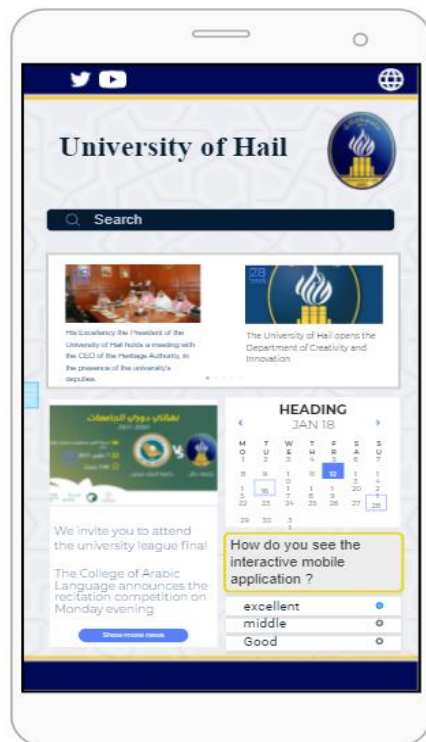
Homepage

Figure 2.11: Homepage.
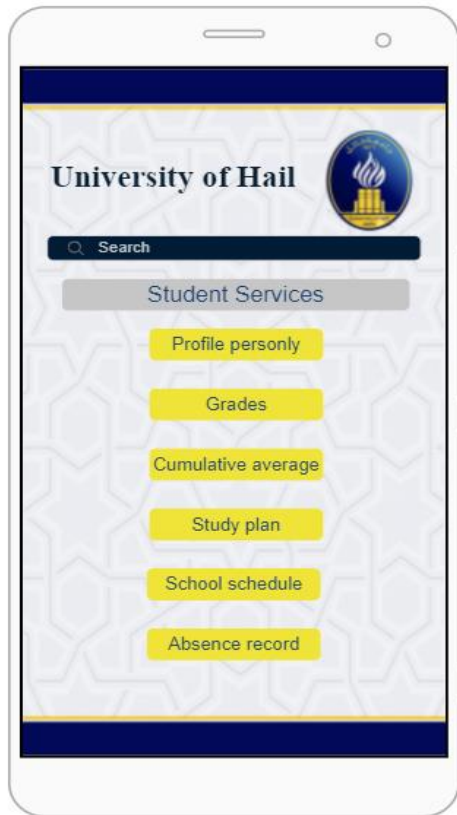
List ofStudent Information

Figure 2.12: List of student services.
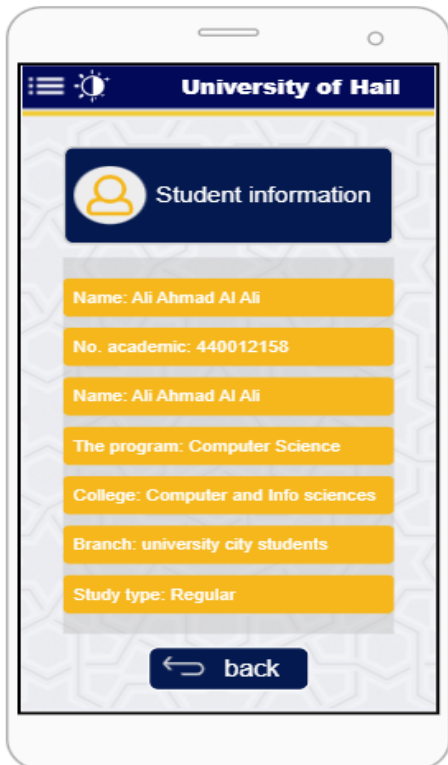
Student Information

Figure 2.13: student information.

3.5 Sequence diagram:

figure 2.14:
sequence diagram
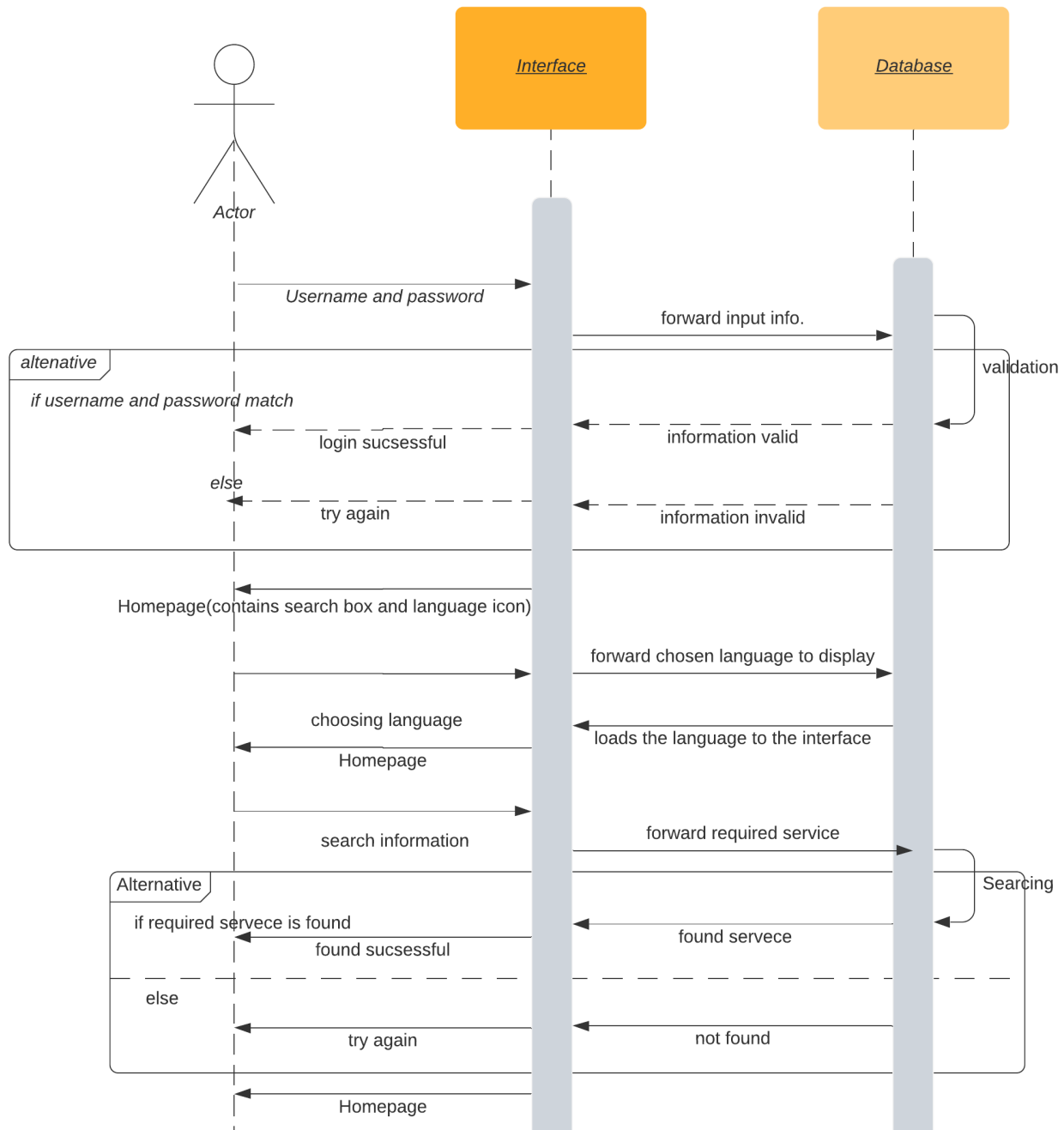
3.Phase 3

Testing

The purpose of this document is to complete the implementation prototype of the project "University Mobile Application for Electronic Services". And validation and verification VV of project functions.
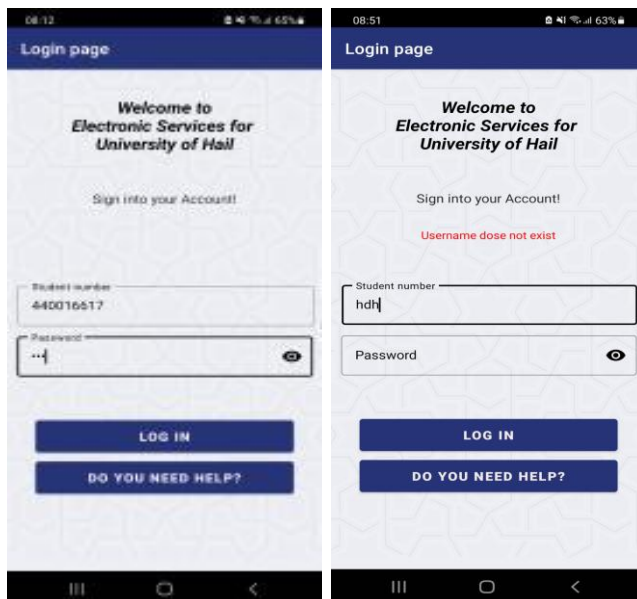
Test Case 1

Test Case ID: Login.

Test Priority: High.

Test Title: Verify login with valid username and password.

Pre-conditions: User has valid username and valid password.

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to the login page.		The system should display a login page.	Login page is displayed.	Pass
2	Provide valid username.	Username = 440016617	The user should be able to enter a username.	The user entered username.	Pass
3	Provide valid password.	Password = ecb	The user should be able to enter a password.	The user entered a password.	Pass
4	Click on the "Login" button.		The system should display the home page.	The system displayed the home page.	Pass

Post-conditions: User is validated with database and successfully login to account. The account session details are logged in the database.
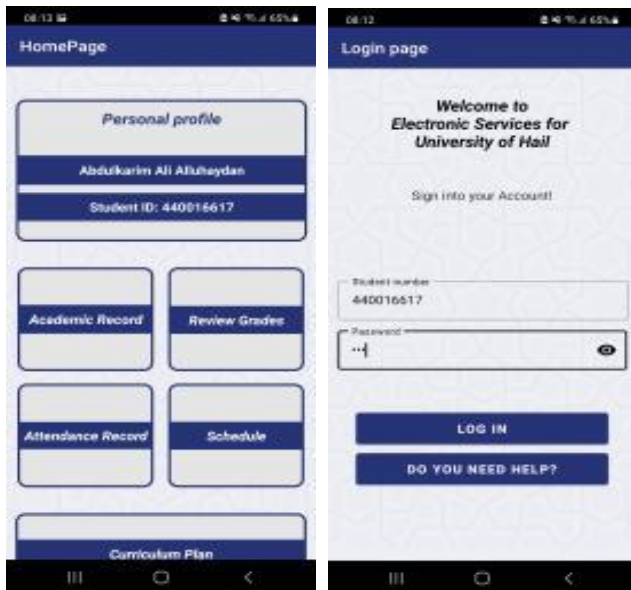
Login Page and Homepage Figures.

Test Case 2

Test Case ID: Login.

Test Priority: High.

Test Title: Verify login with Invalid username and password.

Pre-conditions: User has Invalid username and password.

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to login page		The system should display a login page.	Login page is displayed.	Pass
2	Provide Invalid username.	Username = 440017090	The user should be able to enter a username.	The user entered username.	Pass
3	Provide Password.	Password = ddh1	The user should be able to enter a password.	The user entered password	Pass
4	Click on the "Login" button.		The system should not display home page and display Username does not exist Message	The system did not display home page and displayed Username does not exist Message	Pass

Post-conditions: User is Invalidated with database and did not login to account.

Login Page Figure.

Test Case 3
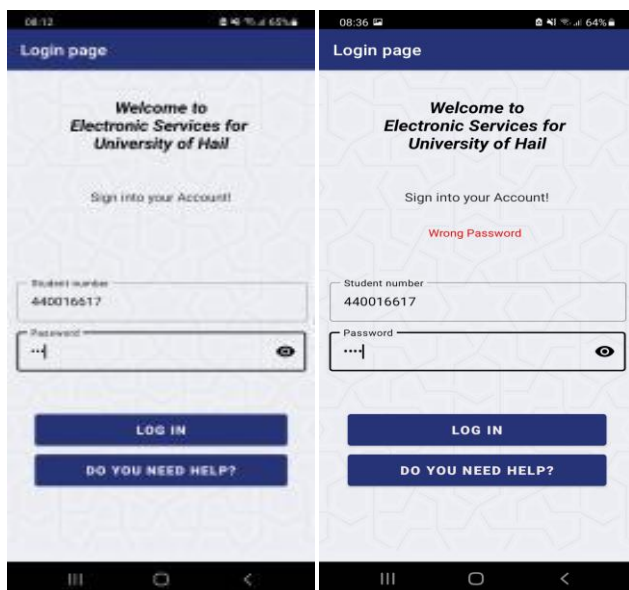
Test Case ID: Login

Test Priority: High

Test Title: Verify login with valid username and Invalid password

Pre-conditions: User has valid username and Invalid password.

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to login page		The system should display a login page.	Login page is displayed.	Pass
2	Provide valid username.	Username = 440016617	The user should be able to enter a username.	The user entered username.	Pass
3	Provide Invalid Password.	Password = ddh1	The user should be able to enter a password.	The user entered password	Pass
4	Click on the "Login" button.		The system should not display home page and display Wrong password Message	The system did not display home page and displayed Wrong password Message	Pass

Post-conditions: User is Invalidated with the database and did not login to account.

Login Page Figure.

Test Case 4

Test Case ID: Personal Profile.

Test Priority: High.

Test Title: Make sure that the user's data is valid and is exactly as it should.

Pre-conditions: User has already defined information such as (name,SID,GPA,etc....)

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to the personal information page from the Home page.	An object of class Student will be passed through.	The system should display the object's attributes as asked.	The system showed the data as expected.	Pass
2	Check the user's name.	getName method will be called.	The user's name should be printed out on the screen.	The user's name is printed.	Pass
3	Check the user's student ID.	getStudentId method will be called.	The user's SID should be printed out on the screen.	The user's SID is printed.	Pass
4	Check the user's phone number.	getPhoneNumber method will be called.	The user's phone number should be printed out on the screen.	The user's phone number is printed.	Pass
5	Check the user's email.	getEmail method will be called.	The user's email should be printed out on the screen.	The user's email is printed.	Pass
6	Check the user's ID.	getId method will be called.	The user's id should be printed out on the screen.	The user's id is printed.	Pass
7	Check the user's age.	getAge method will be called.	The user's age should be printed out on the screen.	The user's age r is printed.	Pass
8	Check the user's salary.	getSalary method will be called.	The user's salary should be printed out on the screen.	The user's salary is printed.	Pass
9	Check the user's major.	getMajor.getName methods will be called.	The user's major should be printed out on the screen.	The user's majors printed.	Pass

Post-conditions: User's information is validated with the database and successfully displayed to the screen.

Personal Profile Page Figure.

Test Case 5

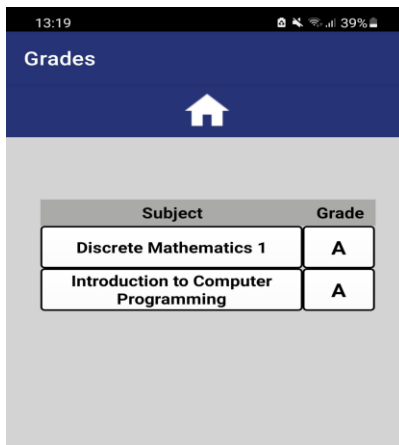**Test Case ID:** Review Grades.

Test Priority: High.

Test Title: Make sure that the user's grade is shown in the correct order, each subject with its final grade.

Pre-conditions: User has already well-defined subjects and grades.

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to the Review Grades page from the Home page.	An array of subjects related to Student will be passed through	The system should display the Review Grades page.	Review Grades is displayed.	Pass
2	Choose the semester.	N semester buttons each related to a semester.	Based on the choice, the system will navigate the user to the right page.	The user is navigated to the wanted page.	Pass
3	Review semester page.	An array of subjects related to Student will be passed through. Based on the semester.	The user's grades will be shown based on how many subjects he has.	The user's grades will be printed out on the screen.	Pass
4	Home Button.	Transfer the user to the homepage.	The system should display the homepage.	The system displayed the homepage.	Pass

Post-conditions: User's grades are validated with the database and are successfully printed out on the screen.

Review Grades Figure.

Test Case 6

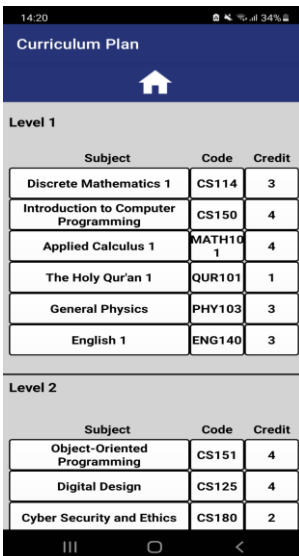**Test Case ID:** Review Curriculum Plan.

Test Priority: High.

Test Title: Ensure that the Curriculum Plan is compatible with the student department.

Pre-conditions: User's department has already been defined.

Step	Test Steps	Test Data	Expected Result	Actual Result	Pass/ Fail
1	Navigate to the Review Curriculum Plan. page from the Home page.	User major will be passed through	The system should display a Curriculum Plan page.	The Curriculum Plan. page is displayed.	Pass
2	Review Curriculum Plan	Major Curriculum Plan get returned	The system should display a Curriculum Plan related to the student department.	Curriculum Plans related to the student department get displayed.	Pass
3	Home Button.	Transfer the user to the homepage.	The system should display the homepage.	The system displayed the homepage.	Pass

Post-conditions: User's department. are validated with the database and department Curriculum Plan are successfully printed out on the screen.

Curriculum Plan Page Figure.