

Kingdom of Saudi Arabia  
Ministry of Education  
Imam Abdulrahman bin Faisal University  
Computer Science department  
College of Science and Humanities

**CS 411 – Software Engineering**  
**Term 1 – 2022/2023**

# **Software Design Specification**

## **For**

**CSHJ**

**Version 1.0**



**CIS Year 4, G5**

**Dr. Norah Alnaim**

**27 October, 2022**

This Software Design Specification was prepared and provided as a deliverable for Software Engineering, CS 411, Term 1 and it will be used by the **CSHJ** team. This document is based in part on the IEEE Recommended Practice for Software Design Descriptions.

## Team Members

#	Student Name	ID
1	<b>May Mohammed AlOtaibi</b>	<b>2200004606</b>
2	<b>Reem Shaker Almualem</b>	<b>2190000429</b>
3	<b>Warood Khalid Alzayer</b>	<b>2190004986</b>
4	<b>Ghala Mohammed Alkhaldi</b>	<b>2200003157</b>
5	<b>Fida Mohammed Alelou</b>	<b>2200003041</b>

## Table of Content

Revision History	4
1. Introduction	8
1.1 Purpose	8
1.2 Scope	8
1.3 Definitions, Acronyms, and Abbreviations	9
1.4 References	11
2. System overview	13
2.1 System Functionality	13
3. Design Considerations	14
3.1 Assumptions and Dependencies	14
3.2 General Constraints	15
4. User Interface Design	16
4.1 Overview of User Interface	16
4.2 Interface Design Rules	18
4.3 Screen Images	19
4.4 Screen Objects and Actions	33
4.5 Other Interfaces	35
5. System Architecture	36
5.1 Architectural Design Approach	36
5.2 Architectural Design	37
5.3 Subsystem Architecture	38
6. Data Design	40
6.1 Data Description	40
6.2 Data Dictionary	44
6.3 Database Description	48
7. Component Design	50
8. Detailed System Design	55
8.1 Classification	55
8.2 Definition	57
8.3 Responsibilities	57
8.4 Constraints	57
8.5 Composition	57
8.6 Uses/Interactions	60
8.7 Resources	68
8.8 Processing	68
8.9 Interface/Exports	75
8.10 Detailed Subsystem Design	80
9. Other Design Features	83
10. Requirements Traceability Matrix	84

## Revision History

Name	Date	Reason For Changes	Version
All members	Oct 17, 2022	Prepared initial version	0.1
All members	Oct 18, 2022	Updated section 3	0.2
All members	Oct 19, 2022	Updated section 4	0.3
All members	Oct 20, 2022	Updated section 5,6	0.4
All members	Oct 21, 2022	Updated section 7	0.5
All members	Oct 22, 2022	Updated section 8	0.6
All members	Oct 23, 2022	Updated section 9,10	0.7
All members	Oct 26, 2022	Complete review - Final version	1.0

## Table of Tables

<b>Table 1</b> Acronyms and definitions.....	11
<b>Table 2</b> Terminologies and definitions .....	12
<b>Table 3</b> users and interfaces.....	19
<b>Table 4</b> screen objects and actions.....	35
<b>Table 5</b> Data Description.....	43
<b>Table 6</b> Data Dictionary.....	47
<b>Table 7</b> Classification.....	57
<b>Table 8</b> Constraints and Composition.....	60
<b>Table 9</b> Resources .....	68
<b>Table 10</b> Rase conditions and Deadlock.....	68
<b>Table 11</b> Login component.....	69
<b>Table 12</b> Login component.....	69
<b>Table 13</b> sign up component.....	69
<b>Table 14</b> change password component.....	70
<b>Table 15</b> forget password component.....	70
<b>Table 16</b> Proper Interface component.....	71
<b>Table 17</b> Edit Profile component.....	71
<b>Table 18</b> join club component.....	72
<b>Table 19</b> raises an objection component.....	72
<b>Table 20</b> review attendance component.....	73
<b>Table 21</b> Verify/Infraction component.....	73
<b>Table 22</b> review attendance/grade component.....	74
<b>Table 23</b> add exam/event schedules component.....	75
<b>Table 24</b> Sign up interface component.....	76
<b>Table 25</b> Login interface component.....	76
<b>Table 26</b> Forget password component.....	77

<b>Table 27</b> Face ID component.....	78
<b>Table 28</b> Join a club.....	78
<b>Table 29</b> Edit profile.....	79
<b>Table 30</b> Requirements Traceability Matrix.....	86

## Table of Figure

Figure 1: User Login.....	19
Figure 2: User sign-up.....	19
Figure 3: Allowing face ID.....	20
Figure 4: Forgot password.....	20
Figure 5: Profile information.....	21
Figure 6: Edit Profile.....	21
Figure 7: Admin's home page.....	22
Figure 8: Add a post.....	22
Figure 9: Admin's club main page.....	23
Figure 10: Add clubs.....	23
Figure 11: Admin's attendance page.....	24
Figure 12:: Admin's attendance page.....	24
Figure 13: Faculty member home page.....	25
Figure 14: Faculty member classes.....	25
Figure 15: Add classes.....	26
Figure 16: Faculty member view attendance.....	26
Figure 17: Security member home page.....	27
Figure 18: Security member violation page.....	27
Figure 19: Add violation.....	28
Figure 20: Security member verification page.....	28
Figure 21: Student home page .....	29
Figure 22: Student clubs.....	29
Figure 23: Student attendance .....	30
Figure 24: Student rules.....	30
Figure 25: Student violations.....	31
Figure 26: System overview.....	36
Figure 27: System layers.....	37
Figure 28: the idea of the interaction with the system.....	37
Figure 29 common functionalities.....	38
Figure 30 Admin's subsystem.....	39
Figure 31 student's subsystem .....	39
Figure 32 CSHJ ERD.....	48
figure 33 ER mapping model.....	49
Figure 34 login functionality.....	60
Figure 35 changing password functionality.....	61
Figure 36 edit profile functionality.....	62
Figure 37 adding a grade to the student.....	63
Figure 38 create and update event information.....	63
Figure 39 giving infraction to the student.....	64
Figure 40 adding the resources to the student.....	64
Figure 41 deleting a resource.....	65
Figure 42 exams to the student.....	66
Figure 43 raise an objection.....	66
Figure 45 viewing attendance.....	67
Figure 46 Joining to clubs.....	67
Figure 47 Flow charts of the system.....	82
Figure 48 the certification pip and filter architecture.....	83

## 1. Introduction

This document, Software Design Specification (SDS) translates all requirements and specifications that have been specified in the previous deliverable, Software Requirement Specification (SRS) into interfaces, architecture, data, and objects that belong to the **CSHJ** artificial intelligence approach-based system. This document describes how the **CSHJ** software will be built and structured to meet the specifications and requirements of the **CSHJ** system. The aim of the system is to provide some features that help the students, faculty members, and administration of the College of Science and Humanities in Jubail in their daily operations. The document has been prepared by following the IEEE Recommended Practice for Software Design Specification (SDS).

SDS will be conducted in two phases[1]:

First phase is **The Initial Design Phase**, which includes designing the interfaces and components of the system that are going to be in the first five sections.

Second phase is **The Final Design Phase**, which includes demonstrating the first phase in detail by using diagrams and models from section six to section ten.

In this section, the purpose of the document as well as the scope, definition, acronyms, and abbreviation will be discussed and defined. Lastly, the references that are used in this document will be addressed.

### 1.1 Purpose

The document will provide detailed information and descriptions of the **CSHJ** system in many aspects such as user and system design, the architecture of the system, and the system database. Also, this document aims to translate the specifications and requirements of the **CSHJ** software onto interfaces, models such as Entity Relationship Model (ERD) and Relational Mapping, and diagrams such as Sequence Diagrams to make an unobstructed vision of the **CSHJ** system architecture[1][2].

### 1.2 Scope

**CSHJ** is an artificial intelligence approach-based system, as well as a system of systems. It is an artificial intelligence approach-based system because it uses the location of the students when they attend the class, as well as verifies the students when they enter the College of Science and Humanities. It is also considered a system of systems because it takes the features of People soft, Blackboard, and My IAU and puts them in one place which is the **CSHJ** application. Furthermore, the users of the system which are students, faculty members, and administration should download the application on their devices in order to take advantage of all features in the application. Because some features work only by downloading the application. This system will manage and ease the workload of the users by gathering the features of many systems in one single system, the users then will not get confused and use many systems.

The users can sign up to the system by using their academic ID or the academic email, and a strong password, and then determine which type of users they are, whether they are students, faculty members, security members, or from the administration. A verification email will be sent to the user's email to verify the account for authentication which will give more security. Moreover, users can use

the touch ID or face ID for more privacy and security. Based on the user type, the user will be redirected to the suitable interface.

#### ❖ The Functionalities of the Administration Interface

There will be a section for adding and deleting students' regulations and rules, as well as a section for adding new events, and clubs' details, a section for receiving official excuses, and a section for sending notifications to students and other users which include the admins, faculty members, and security members.

#### ❖ The Functionalities of the Faculty Members Interface

For the faculty members, there will be a section for reviewing the attendance as well as the absence of the students after the automatic attendance, also a section for adding and deleting notifications and classes schedule and times. Moreover, a section for providing students with certificates of thanks through the platform.

#### ❖ The Functionalities of the Security Members Interface

For the security members, there will be a section for giving the students infractions when they violate the rules, as well as ensuring that the students from the college.

#### ❖ The Functionalities of Students Interface

For the students, there will be a section for reviewing the attendance for each course, they will be able to inform the faculty members if they will not attend the class through a chatbot. Students will be able to submit an official excuse through this platform. Also, there will be a section for evaluating the faculty members after the classes as well as there will be a section for taking notes in the class and they will be saved automatically. Through this platform, students will be able to raise an objection to any violation. In the section related to the clubs, students will be able to enroll in any of the college clubs.

For all users, there will be a section called a notification stream, which includes the most recent news of the college, as well as the events and any other announcements related to the College of Science and Humanities.

In this document, more details about **CSHJ** will be covered. With these details, the implementation stage will be facilitated by using the diagrams and the models to explain the architecture and the flow of the data in the **CSHJ** platform.

### 1.3 Definitions, Acronyms, and Abbreviations

<b>Acronyms</b>	<b>Definitions</b>
<b>CSHJ</b>	<b>College of Science and Humanities in Jubail</b>
<b>ID</b>	<b>Identification</b>

<b>SRS</b>	<b>Software Requirements Specification</b>
<b>SDS</b>	<b>Software Design Specification</b>
<b>IEEE</b>	<b>The Institute of Electrical and Electronic Engineering</b>
<b>IAU</b>	<b>Imam Abdulrahman Bin Faisal University</b>
<b>ERD</b>	<b>Relational Entity Diagram</b>
<b>SQL</b>	<b>Structure Query Language</b>
<b>XAMPP</b>	<b>X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and the Ps stand for PHP and Perl</b>
<b>GUI</b>	<b>Graphical User Interface</b>
<b>JVM</b>	<b>Java Virtual Machine</b>

Table 1 Acronyms and definitions

<b><i>Terminology</i></b>	<b><i>Definitions</i></b>
<b>Interface</b>	<b>Is a group of related methods with empty bodies[3]0</b>
<b>Software</b>	<b>Is a set of instructions, data, or programs used to operate computers and execute specific tasks. [4]0</b>
<b>Database</b>	<b>Organized collection of structured information.0</b>

<b>User</b>	<b>The person, or persons, who operate or interact directly with the product. The user(s) and the customer(s) are often not the same people[1]0</b>
<b>Systems of Systems</b>	<b>Consist of several different software systems[1] 0</b>
<b>Artificial Intelligence</b>	<b>A significant field of computer science is concerned with creating intelligent machines capable of doing activities that normally need human intellect[5]0</b>
<b>Entity Relation Diagram</b>	<b>A diagrams that demonstrates the entities and attributes with their relationship [2]</b>
<b>Entity</b>	<b>Basic concept of the ER model[2].</b>
<b>Attributes</b>	<b>Properties used to describe the entity[2].</b>
<b>MySQL</b>	<b>It is a database management system (open-source) [2]</b>
<b>Entity Relation Model</b>	<b>A model that demonstrates the entities and attributes with their relationship [2]</b>

Table 2 Terminologies and definitions

#### 1.4 References

[1]Somerville, Ian, “Software Engineering”, Addison Wesley (10th edition): 2016, ISBN:10:1-292-09613-6, ISBN-13:978-1-292-09613-1.

[2]Elmasri, R. and Navathe, S. (2020) *Fundamentals of Database Systems*. Hoboken, NJ: Pearson.

[3](Oracle, 1995,2021)

[4] (SearchAppArchitecture, 2021)

[5] Burns, E., Laskowski, N., & Tucci, L. (2021, July 1). What is Artificial Intelligence (AI)? - ai definition and how it works. SearchEnterpriseAI. Retrieved October 4, 2021, from <https://searchenterpriseai.techtarget.com/definition/AI-Artificial-Intelligence>.

[6]] "IEEE Recommended Practice for Software Design Specifications," in IEEE Std 1016-1998 , vol., no., pp.1-40, 20 Oct. 1998.

- [7] Blackboard, “E-Learning Management System,” *Blackboard Learn*. [Online]. Available: <https://vle.iau.edu.sa/>. [Accessed: 27-Oct-2022].
- [8] “twitter,” *Twitter*. [Online]. Available: <https://twitter.com/?lang=ar>. [Accessed: 27-Oct-2022].
- [9] I. A. B. F. University, “My iau,” *App Store*, 15-Mar-2017. [Online]. Available: <https://apps.apple.com/us/app/my-iau/id1214071868>. [Accessed: 27-Oct-2022].
- [10] J. TyusStudent, J. Tyus, Student, and S. University, “Blackboard app: Mobile Learning Solutions,” *Blackboard*. [Online]. Available: <https://www.blackboard.com/teaching-learning/learning-management/mobile-learning-solutions>. [Accessed: 27-Oct-2022].

## 2. System overview

**CSHJ** is designed to gather all faculty members' and students of the College of Science and Humanities in Jubail in one interactive place. It provides many services that facilitate what the faculty members and students face on a daily basis like: student attendance in classes, student ID verification at the college entrance, view latest classes for students, display and register in all student clubs within the college. As well as display the violations committed by the student inside the university campus and finally lack of one single place that gathers all college notifications such as most recent events, workshops, dress code, and other activities that both administration and faculty members would like to add. This project takes some features from both blackboard, people soft and my IAU application with more improved features and technologies that keep up with the rapid changes nowadays. In this section the overall description of the system functionality will be provided. The functions that will be performed are listed below ( Mentioned in detail previously in the SRS document):

### 2.1 System Functionality

- ❖ Registration: All end users have the ability to register into the system.
- ❖ Log-in: After registration, the end users can log-in easily using their college academic email and password.
- ❖ Verification: End users can confirm their identity by allowing face recognition features. This will make it easier when trying to enter the system again.
- ❖ Account setting: The end users will be able to edit and modify the information of the user account.
- ❖ Language: End users can choose between Arabic or English for the language of the application.
- ❖ Viewing the information: End users will be able to view all information shared on the streaming area.
- ❖ Deletion: The administration has the ability to delete any post.
- ❖ Editing: The administration has the ability to edit any post, clubs and classes and attendance.
- ❖ Posting: The administration, faculty members have the ability to post in the streaming area.
- ❖ Student verification: Student ID verification will be done automatically using the attendance technology.
- ❖ Joining: Students can enroll themselves into student clubs.
- ❖ Attendance: students can view their attendance schedule and details.
- ❖ Managing attendance: Admins can accept or reject the official excuses.
- ❖ Class schedule: A faculty member can schedule their own classes and lectures.

### 3. Design Considerations

This section focuses on examining the "**CSHJ**" assumptions and dependencies, as well as the general constraints that might impact the system.

#### 3.1 Assumptions and Dependencies

The software's assumptions or dependencies will be categorized as follows in this section:

- Related software or hardware.
- Operating systems.
- End-user characteristics.
- Possible and/or probable changes in functionality.

##### 3.1.1 Related Software or Hardware

The **CSHJ** system can be accessed with any pc, tablet, or mobile device. The web-based nature of the software involves a connection to the internet. In addition to the system-related software, such as MySQL, a straightforward database architecture should be created in order to prevent data loss and protect the system's information. The database should prepare for simultaneous updates being made by a variety of users who have access to the system. The system is developed and tested using the NetBeans IDE as well. The system needed a server computer with a high-band internet connection, per the hardware. To ensure continuous availability, the server machine will feature a strong CPU and high-speed internet connectivity.

##### 3.1.2 Operating System

iOS, Android, and Windows 10 operating systems will all be suitable for the "**CSHJ**" system. Windows 10 and iOS will be used by the developer team as two independent platforms.

##### 3.1.3 End-user characteristics

**CSHJ** will have four types of users, which are administration, faculty members, security members, and Students. The administration can add and delete students' regulations and rules, as well as a section for adding new events, and clubs' details, and send notifications to students and other users which include the admins, faculty members, and security members. Faculty members can review the attendance as well as the absence of the students after the automatic attendance and add and delete notifications and classes schedule and times. Security members are able to give the students infractions when they violate the rules, as well as ensure that the students from the college. Students reviewing their attendance for each course, can submit an official excuse through this platform, evaluate the faculty members after the classes, and join the clubs.

### 3.1.4 Possible and/or probable changes in functionality General Constraints

It's possible that some functionality will change and that the final design will be different from the initial one. Future updates to the platform could also include certain feature additions. Only small alterations may be made since the project must be completed on time; otherwise, the deadlines and schedules will be affected.

## 3.2 General Constraints

This section presents the general constraints that affect the software design, which are:

- Hardware or software environment.
- Interface/protocol requirements.
- Data repository and distribution requirements.
- Security requirements (or other such regulations).
- Performance requirements.
- Network communications.
- Verification and validation requirements (testing).

### 3.2.1 Hardware or software environment

The platform will be accessible with any browser and will run on both Android and iOS, as detailed in the SPMP document. Developers will need MacOS or Windows 10 to work and MySQL Workbench to store and retrieve data, they will also require the NetBeans IDE.

### 3.2.2 Interface/protocol requirements

**CSHJ** features user-friendly interfaces that are beneficial to users. The **CSHJ** interface has a significant impact, including the following:

- Simplicity.
- Desirability.
- Clarity.
- Accessibility.

### 3.2.3 Data repository and distribution requirements

The MySQL Workbench will be used to store the data in a database. Additionally, the data will be retrieved using MySQL Workbench.

### 3.2.4 Security requirements

Users should be certain to preserve their personal data in secure storage. An academic email will be required for registration on the application. In addition, choose a strong password.

### 3.2.5 Performance requirements

Every criterion listed previously must be tested when the user accesses the platform. In the event that one of the conditions is not met, any additional modifications will be processed appropriately and altered.

### 3.2.6 Network communication

**CSHJ** must be entered through a network, similar to any other website. The website simply cannot function otherwise. To connect and deploy the **CSHJ** website into the network using XAMAPP, a server is required.

### 3.2.7 Verification and validation requirements (testing)

Testing, spotting, and debugging errors that conflict with platform requirements are all goals of the V&V. The incremental model is employed in this project, as stated in the SPMP document (Agile Model). In order to provide feedback, the customer must compare the platform to the necessary requirements.

## 4. User Interface Design

### 4.1 Overview of User Interface

**CSHJ** system has a user-friendly interfaces that provides acceptability[1] that means all interfaces must be easy and understandable to the users to use. In this system there are four types of users which are the administration members, faculty members, security members, and students. Every user has a different interface depending on the functions that they need.

## Common Interfaces

- Home page
- Sign-up
- Forgot password
- Allowing face ID
- Profile information
- Edit profile

User	Interfaces
<b>Administration</b>	<ul style="list-style-type: none"><li>- Admin's home page</li><li>- Add a post</li><li>- Admin's club main page</li><li>- Add clubs</li><li>- Admin's Attendance page</li><li>- Admin's rules page</li></ul>
<b>Faculty members</b>	<ul style="list-style-type: none"><li>- home page</li><li>- Classes page</li><li>- Add a class</li><li>- View Attendance</li></ul>

<b>Security members</b>	<ul style="list-style-type: none"> <li>- home page</li> <li>- Violation page</li> <li>- Add violation</li> <li>- Verification page</li> </ul>
<b>Students</b>	<ul style="list-style-type: none"> <li>- Home page</li> <li>- Clubs page</li> <li>- Attendance</li> <li>- Rules streaming page</li> <li>- Violation page</li> </ul>

Table 3 users and interfaces

## 4.2 Interface Design Rules

The **CSHJ** Follows the principle of Eight Golden Rules to achieve the acceptability and provide clear, understandable, easy to use for all users interfaces. The rules are:

- ★ **Strive for consistency:** the interfaces must be consistent which means it should use consistent fonts, layout, and colors.
- ★ **Enable frequent users to use shortcuts:** this helps users, especially the advanced ones to perform quick action and by less effort.
- ★ **Offer informative feedback:** it is important to provide the users with feedback so they know what is going on. It must be clear what is going to be the response of the system while designing the interfaces.
- ★ **Design dialogs to yield closure:** the users should be informed if the task is completed or not. A dialog box or a simple message can help the users understand what is going on and know the consequences of the actions that they do.
- ★ **Prevent errors:** if an error has occurred, the system must handle it as painlessly as possible.
- ★ **Permit easy reversal of actions:** users should be able to undo or redo their action.
- ★ **Keep users in control :** the system must be designed in a way that it gives the users some degree of control.
- ★ **Reduce short-term memory load:** the system interfaces must be designed in a way that the information should be preserved and the user is not required to recall the information from previous interfaces.

### 4.3 Screen Images

This section demonstrates the available interfaces and the functionality for each user. The designing phase of all the interfaces are inspired from blackboard (the website), blackboard (the application), my iau application and finally twitter [7-10].

#### 4.3.1 Common Interfaces



Email  
2190004986@iau.edu.sa

PASSWORD  
\*\*\*\*\*

**LOG IN**

[FORGOT PASSWORD?](#)  
[SIGNUP](#)

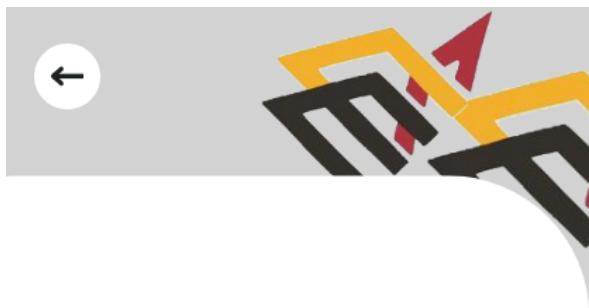
The sign-up interface has a header "Create New Account" with a back arrow icon. Below it, a link "Already Registered? Log in here." is shown. The form fields include:

- NAME: Warood Khalid
- ACADEMIC EMAIL: 2190004986@iau.edu.sa
- PASSWORD: \*\*\*\*\*  
Password must be at least 8 character, contain number and capital
- COLLEGE NAME: Select
- POSITION: Select

**sign Up**

Figure 1: User Login

Figure 2: User sign-up



**Allow Face ID**



**DO YOU WANT TO ALLOW  
"CSHJ" TO USE FACE ID?**

CSHJ USES FACE ID TO RESTRICT  
UNAUTHORIZED USERS FROM  
ACCESSING THE APP

Don't Allow

ok

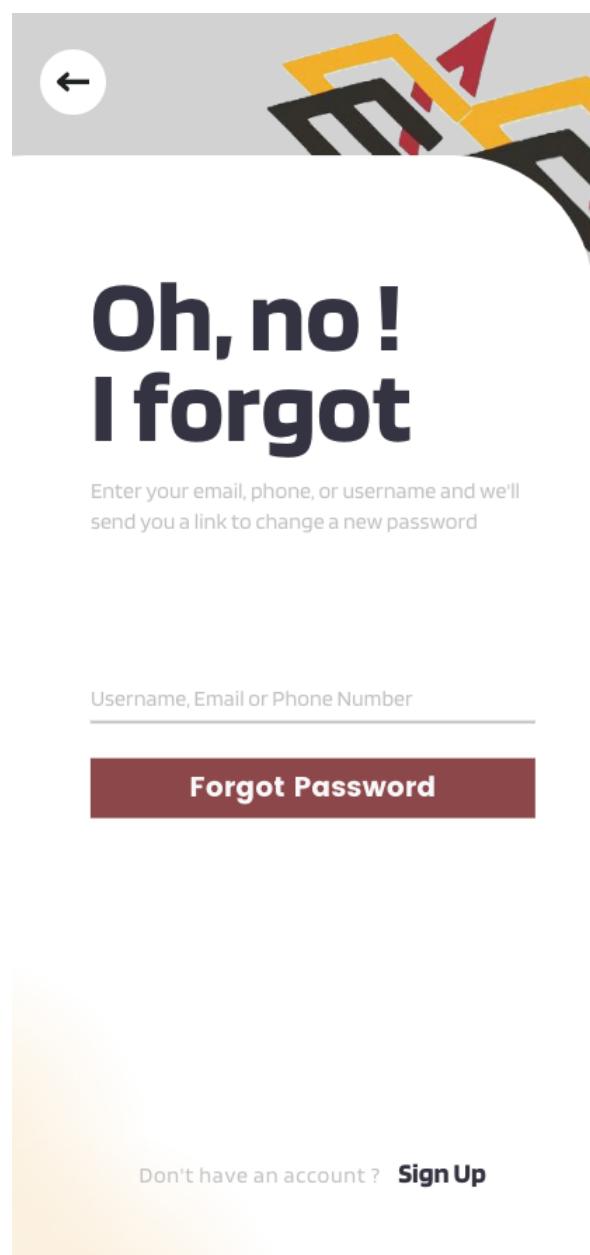


Figure 3: Allowing face ID

Figure 4: Forgot password

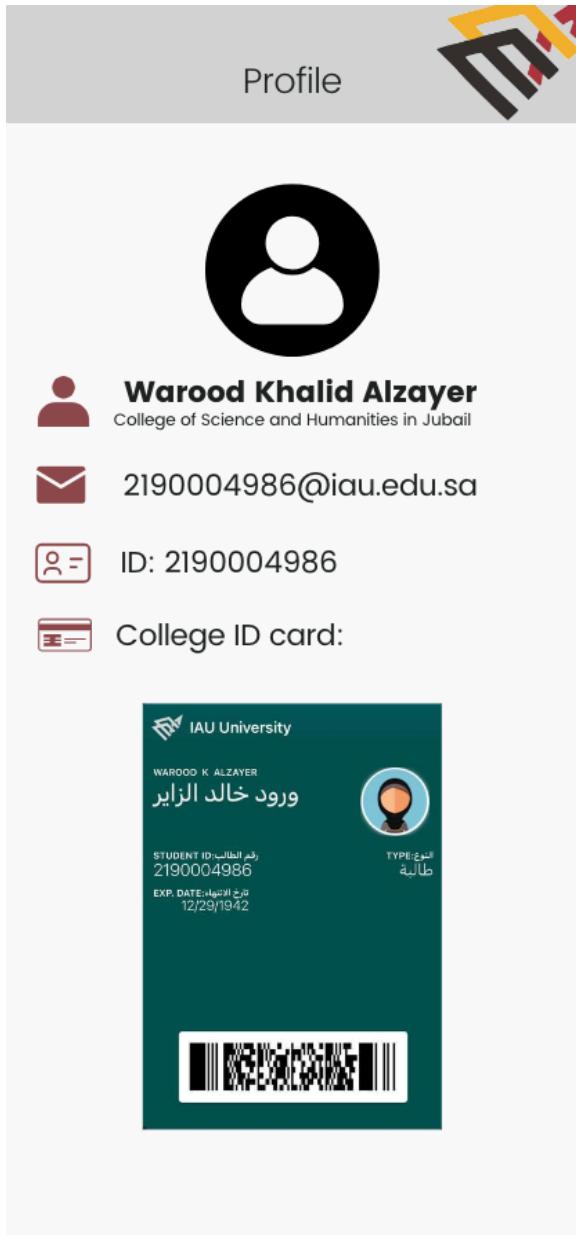


Figure 5: Profile information

The screenshot shows the "Edit Profile" screen. At the top right is a back arrow icon. The title "Edit Profile" is centered below it. The form consists of several input fields: "NAME" with the value "Warood Khalid", "ACADEMIC EMAIL" with the value "2190004986@iau.edu.sa", "COLLEGE NAME" with the value "Select", and "POSITION" with the value "Select". At the bottom right is a large red "Done" button.

Figure 6: Edit Profile

### 4.3.2 Administration Interfaces

The screenshot shows the Admin's home page with the following content:

- Activity Stream:**
  - Administration:** Dress Code: long pants or skirts.
  - Reem Shaker:** 219000412 calculus girls don't forget to bring your laptops
  - Dr. Norah Alnaim:** Faculty Member. There will be a workshop about: introduction to cyber security in the college theater
  - Dr. Maryam Khalid:** Faculty Member. for those who want to register in student sport activities please go to office #25
  - Administration:** Admin. class # 145 A will be closed until further notice
  - Dr. Sara Turkey:** Faculty Member. if you want to register in the computer science club please
- Bottom navigation:** Stream, Clubs, Attendance, Rules

Figure 7: Admin's home page

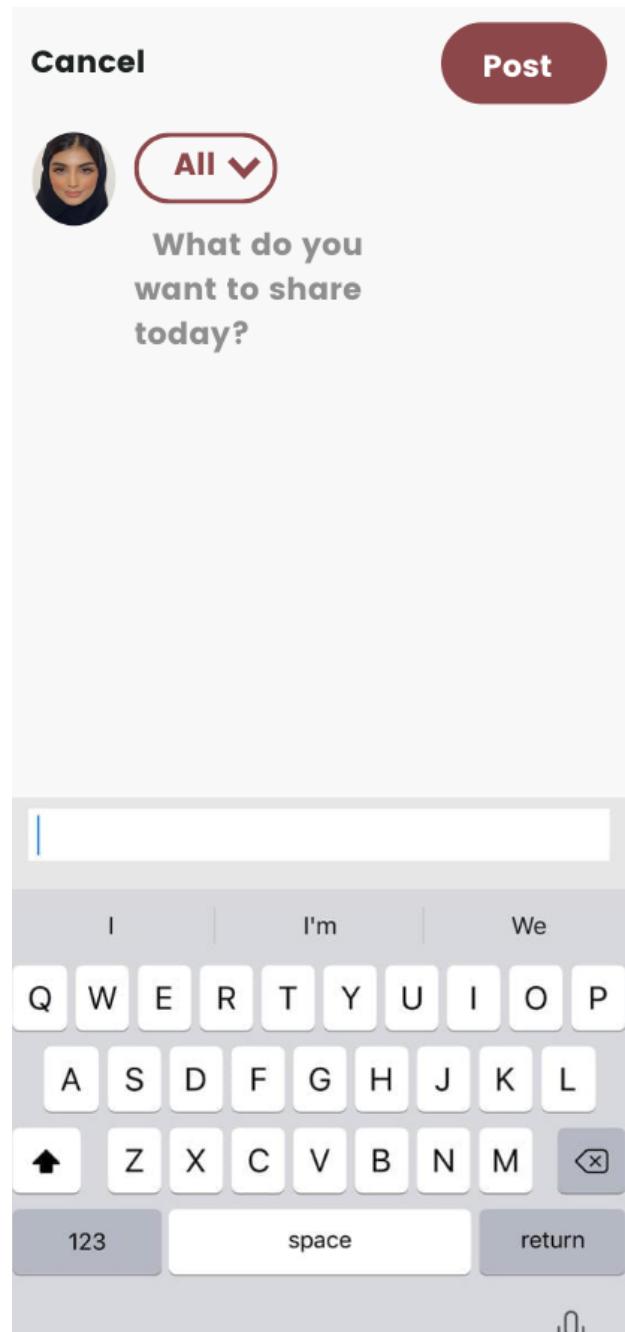


Figure 8: Add a post

**Student Clubs**

**Computer Science Club** ...  
Our mission is to grow and strengthen the student developer community here at Pitt and connect our members to all the people, resources, and opportunities they want to achieve their goals.

**Art Club** ...  
the Arts Club is home to a community of discerning thinkers and thought leaders, with a shared appreciation for the arts and culture.

**+**

**Stream** **Clubs** **Attendance** **Rules**

Figure 9: Admin's club main page

**Add a Student Club**

**NAME**  
Physics club

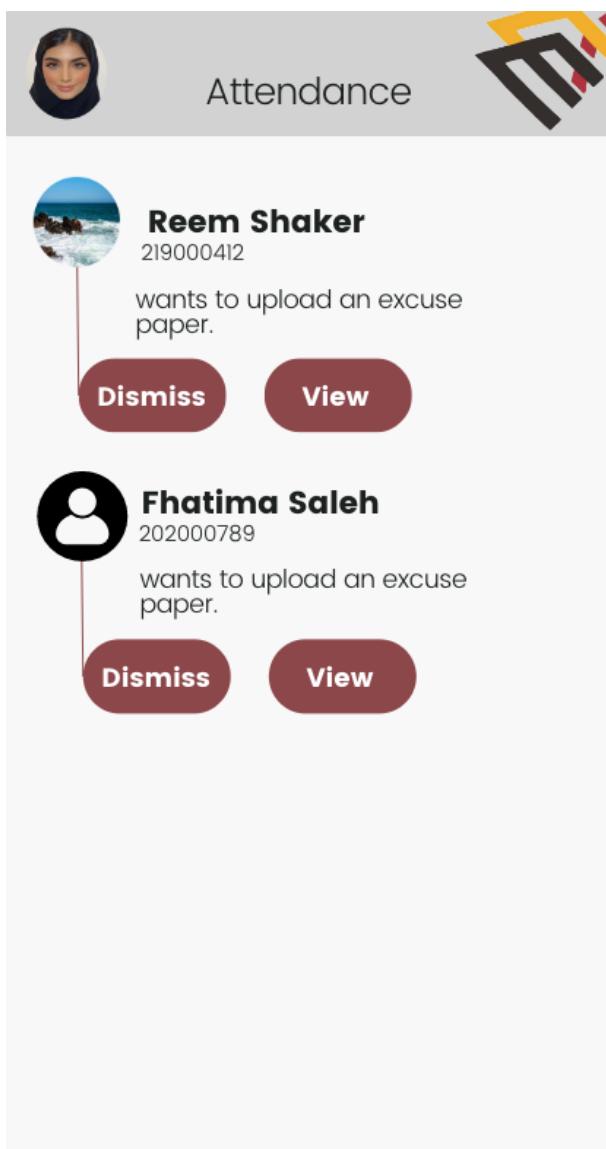
**DESCRIPTION**  
Physics Club consists of students from the Department of Physics who endeavor to keep up with the new trends in the world of Physics.

**NUMBER OF STUDENTS ALLOWED**  
Select

**ADD A PICTURE**  
Upload

**Add**

Figure 10: Add clubs

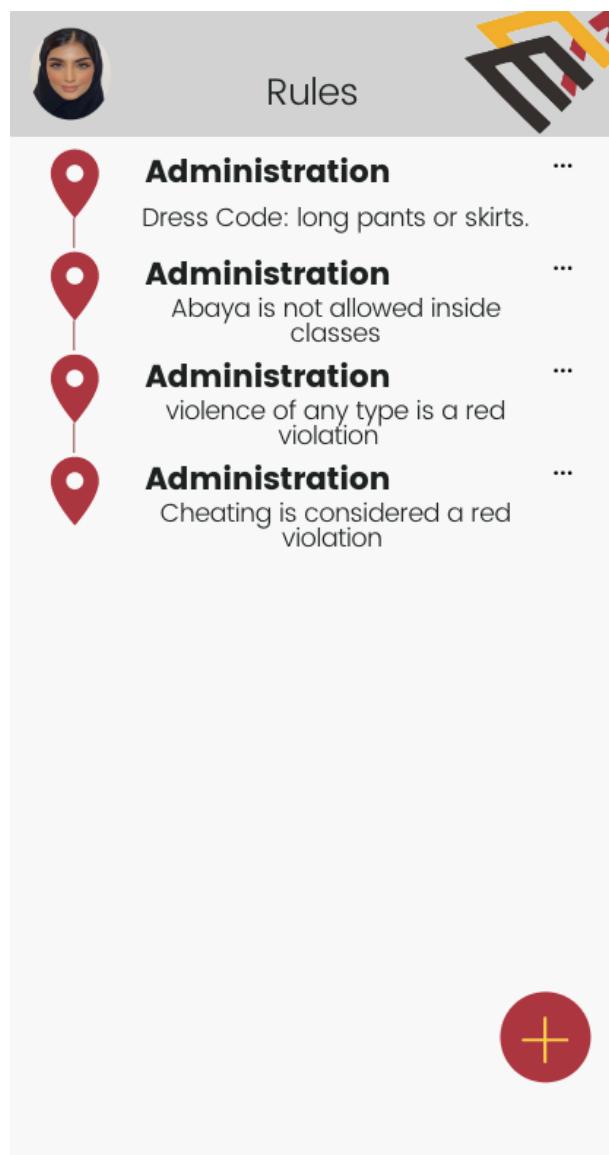


The screenshot shows the 'Attendance' section of the admin interface. At the top, there is a placeholder profile picture and the word 'Attendance' next to a stylized 'M' logo. Below this, two student entries are listed:

- Reem Shaker**, ID 219000412, wants to upload an excuse paper. Includes a circular profile picture of a beach scene, a 'Dismiss' button, and a 'View' button.
- Fhatima Saleh**, ID 202000789, wants to upload an excuse paper. Includes a circular profile picture of a person, a 'Dismiss' button, and a 'View' button.

At the bottom, there are four navigation icons: Stream (audio波形), Clubs (people icon), Attendance (calendar icon), and Rules (gavel icon).

Figure 11: Admin's attendance page



The screenshot shows the 'Rules' section of the admin interface. At the top, there is a placeholder profile picture and the word 'Rules' next to a stylized 'M' logo. Below this, four red location pin icons are followed by 'Administration' sections with specific rules:

- Dress Code: long pants or skirts.
- Abaya is not allowed inside classes.
- violence of any type is a red violation.
- Cheating is considered a red violation.

At the bottom right is a large red circle with a white plus sign. At the very bottom, there are four navigation icons: Stream (audio波形), Clubs (people icon), Attendance (calendar icon), and Rules (gavel icon).

Figure 12: Admin's rules page

### 4.3.3 Faculty Members Interfaces

The interface is titled "Activity Stream" and features a profile picture of a female faculty member. It displays several posts from different users:

- Administration**: Dress Code: long pants or skirts.
- Reem Shaker**: calculus girls don't forget to bring your laptops
- Dr. Norah Alnaim**: Faculty Member. There will be a workshop about: introduction to cyber security in the college theater
- Dr. Maryam Khalid**: Faculty Member. for those who want to register in student sport activities please go to office #25
- Administration**: Admin. class # 145 A will be closed until further notice
- Dr. Sara Turkey**: Faculty Member. if you want to register in the computer science club please

At the bottom, there are three navigation icons: Stream (audio wave), Classes (two people), and Attendance (calendar).

Figure 13: Faculty member home page

The interface is titled "Classes" and shows a post for the "Software Engineering" class:

- Software Engineering**

Below the title are two buttons: "Edit" and "View". At the bottom right is a red circular button with a plus sign. The bottom navigation bar includes Stream, Classes, and Attendance icons.

Figure 14: Faculty member classes



## Add a Class

NAME

Artificial Intelligence

DESCRIPTION

learn what AI is, what can (and can't) be done with AI, and how to start creating AI methods. The courses combine theory with practical exercises and can be completed at your own pace.

DAYS

Select

TIME

Select

HALL

I45A

**Add**

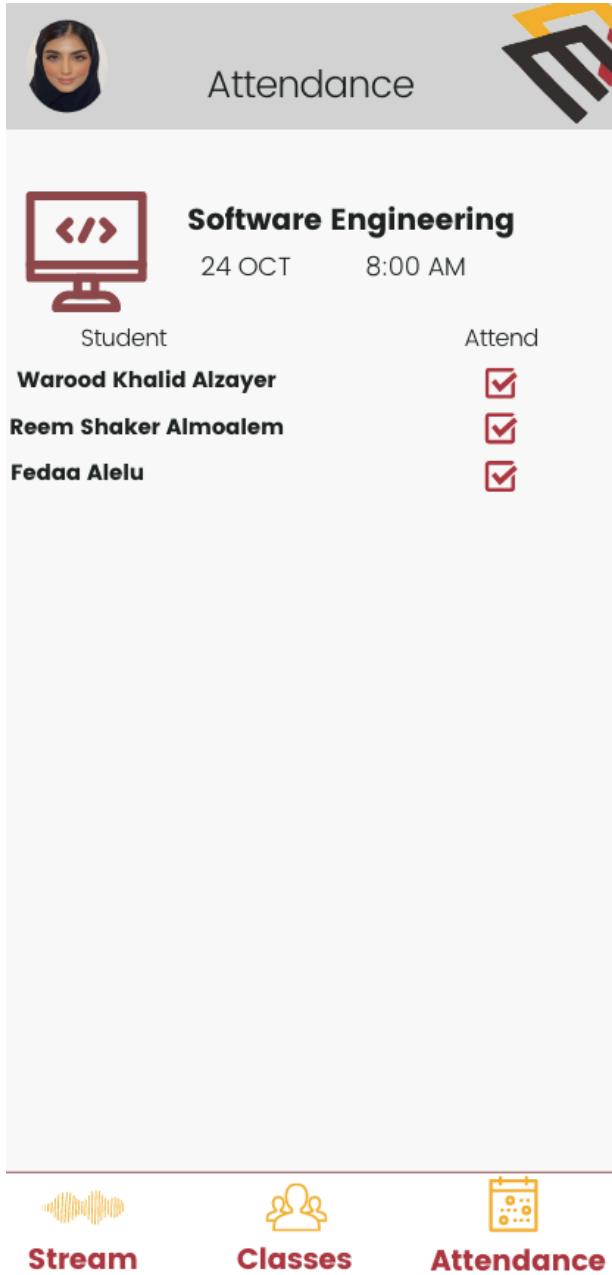


Figure 15: Add classes

Figure 16: Faculty member view attendance

#### 4.3.4 Security Members Interfaces

The screenshot shows the 'Activity Stream' section of the security member interface. It displays several posts from different users:

- Administration**: Dress Code: long pants or skirts.
- Reem Shaker**: calculus girls don't forget to bring your laptops
- Dr. Norah Alnaim**: Faculty Member. There will be a workshop about: introduction to cyber security in the college theater
- Dr. Maryam Khalid**: Faculty Member. for those who want to register in student sport activities please go to office #25
- Administration**: Admin. class # 145 A will be closed until further notice
- Dr. Sara Turkey**: Faculty Member. if you want to register in the computer science club please

At the bottom, there are three navigation icons: Stream (audio wave icon), Violations (red X icon), and verification (yellow person icon).

Figure 17: Security member home page

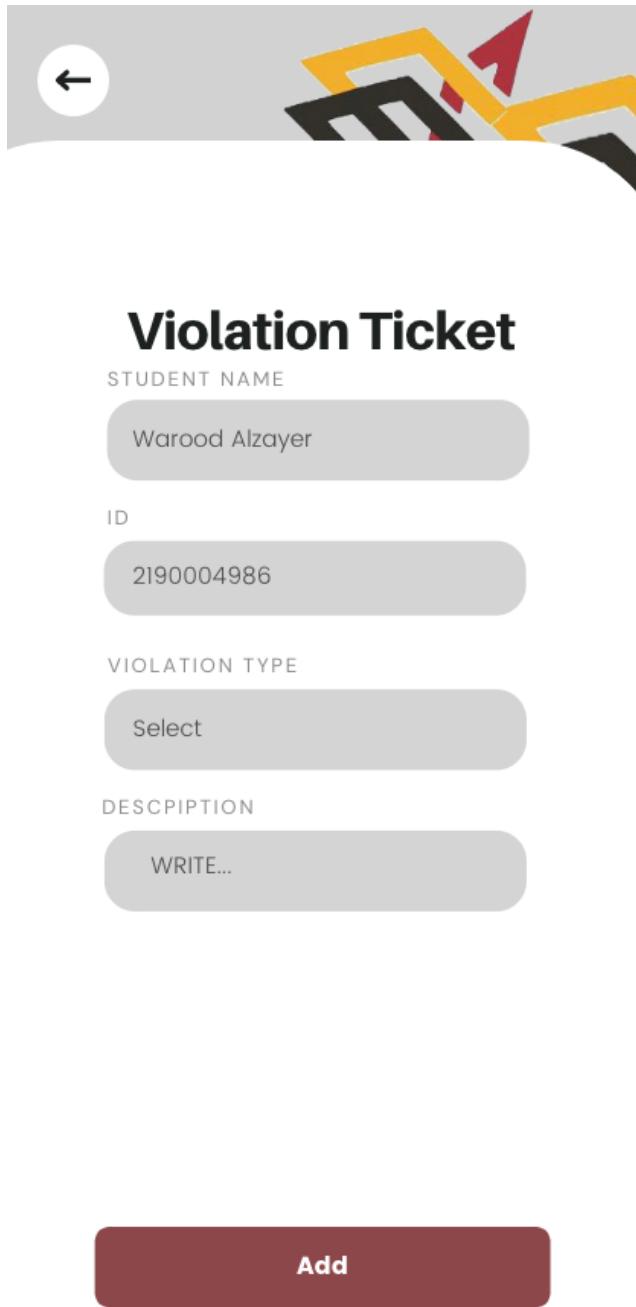
The screenshot shows the 'Violations' section of the security member interface. It displays a list of violations:

- Violations

Below the list is a large graphic showing two overlapping black L-shaped brackets with a central plus sign, representing a barcode scanning area. Below this graphic is the text: "Scan student barcode to free a violation".

At the bottom, there are three navigation icons: Stream (audio wave icon), Violations (red X icon), and verification (yellow person icon).

Figure 18: Security member violation page



The screenshot shows a mobile application interface for adding a violation ticket. At the top left is a back arrow icon. To the right is a stylized logo consisting of overlapping yellow, red, and black geometric shapes. Below the logo, the title "Violation Ticket" is displayed in large, bold, black font. Underneath the title, the label "STUDENT NAME" is followed by a text input field containing "Warood Alzayer". The next section is labeled "ID" with a text input field containing "2190004986". Below these is a section labeled "VIOLATION TYPE" with a text input field containing "Select". Under "DESCRIPTION", there is a placeholder text input field with "WRITE...". At the bottom is a large, dark red button with the word "Add" in white.

Figure 19: Add violation

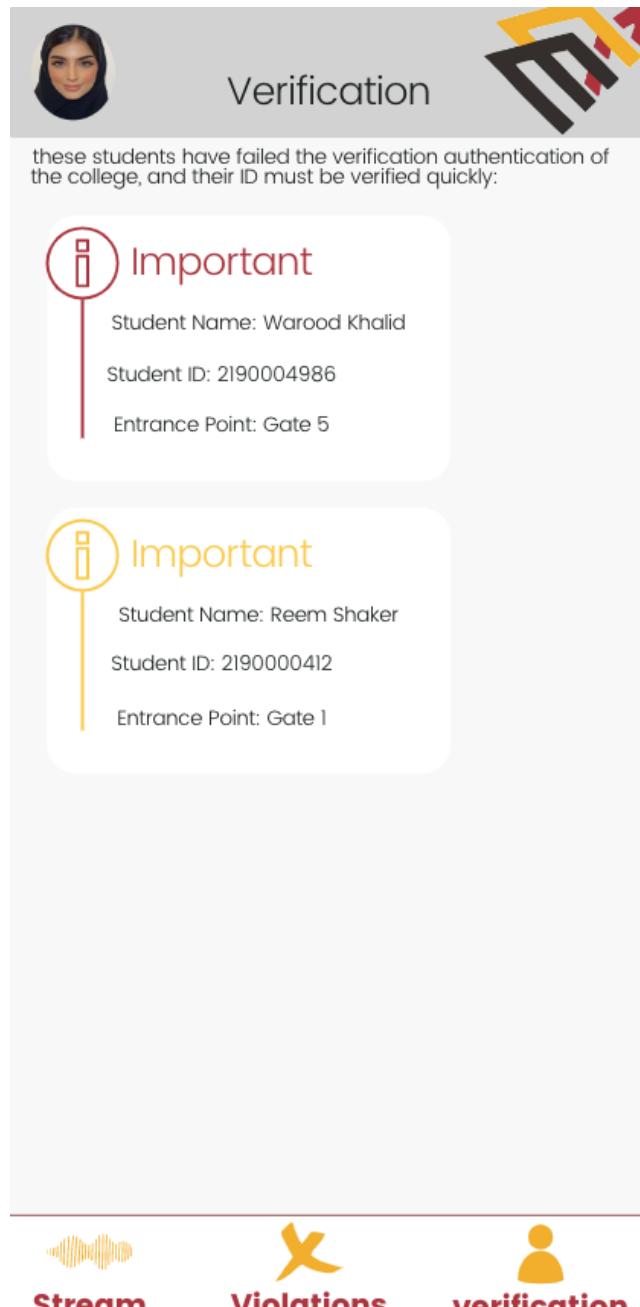


Figure 20: Security member verification page

### 4.3.5 Student Interfaces

The screenshot shows the 'Activity Stream' section of the student interface. It features a profile picture of a woman at the top left. Below it, there are several posts from different users:

- Administration**: Dress Code: long pants or skirts.
- Reem Shaker**: 219000412. calculus girls don't forget to bring your laptops
- Dr. Norah Alnaim**: Faculty Member. There will be a workshop about: introduction to cyber security in the college theater
- Dr. Maryam Khalid**: Faculty Member. for those who want to register in student sport activities please go to office #25
- Administration**: Admin. class # 145 A will be closed until further notice
- Dr. Sara Turkey**: Faculty Member. if you want to register in the computer science club please

At the bottom, there are four navigation icons: Stream (audio波形), Clubs (人群), Attendance (日历), and More (三个圆点) with a red plus sign.

Figure 21: Student home page

The screenshot shows the 'Student Clubs' section. It features a profile picture of a woman at the top left. Below it, there are two club profiles:

- Computer Science Club**: Our mission is to grow and strengthen the student developer community here at Pitt and connect our members to all the people, resources, and opportunities they want to achieve their goals.
- Art Club**: the Arts Club is home to a community of discerning thinkers and thought leaders, with a shared appreciation for the arts and culture.

Each club profile has a red circular icon with a yellow plus sign to its right. At the bottom, there are four navigation icons: Stream (audio波形), Clubs (人群), Attendance (日历), and More (三个圆点) with a red plus sign.

Figure 22: Student clubs

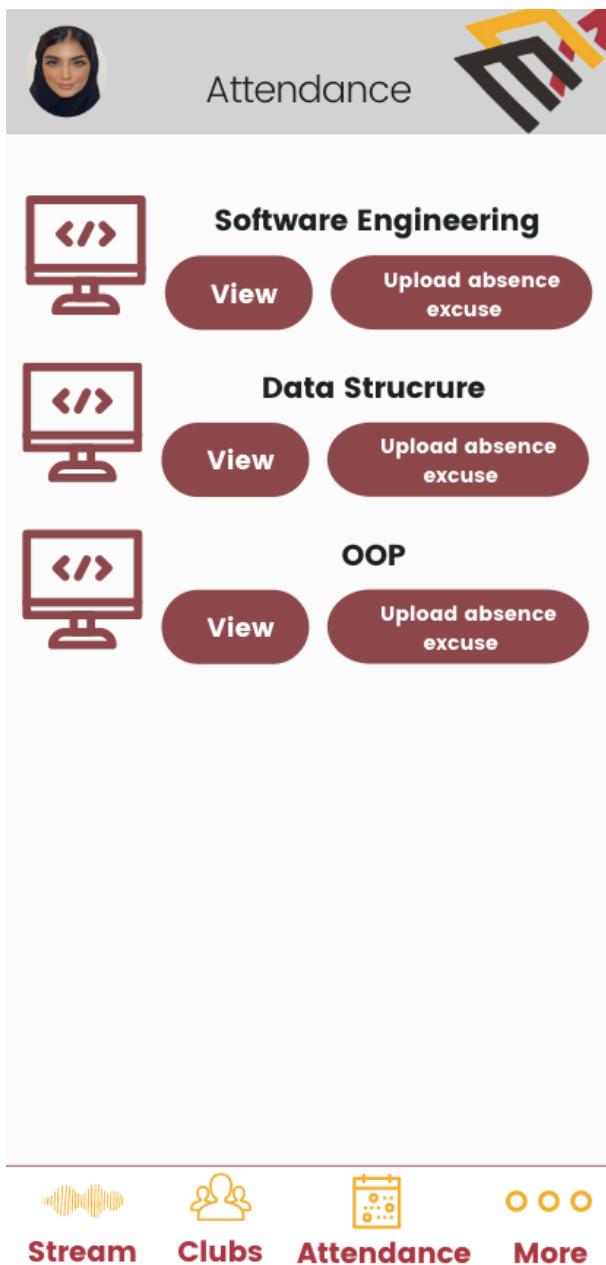


Figure 23: Student attendance

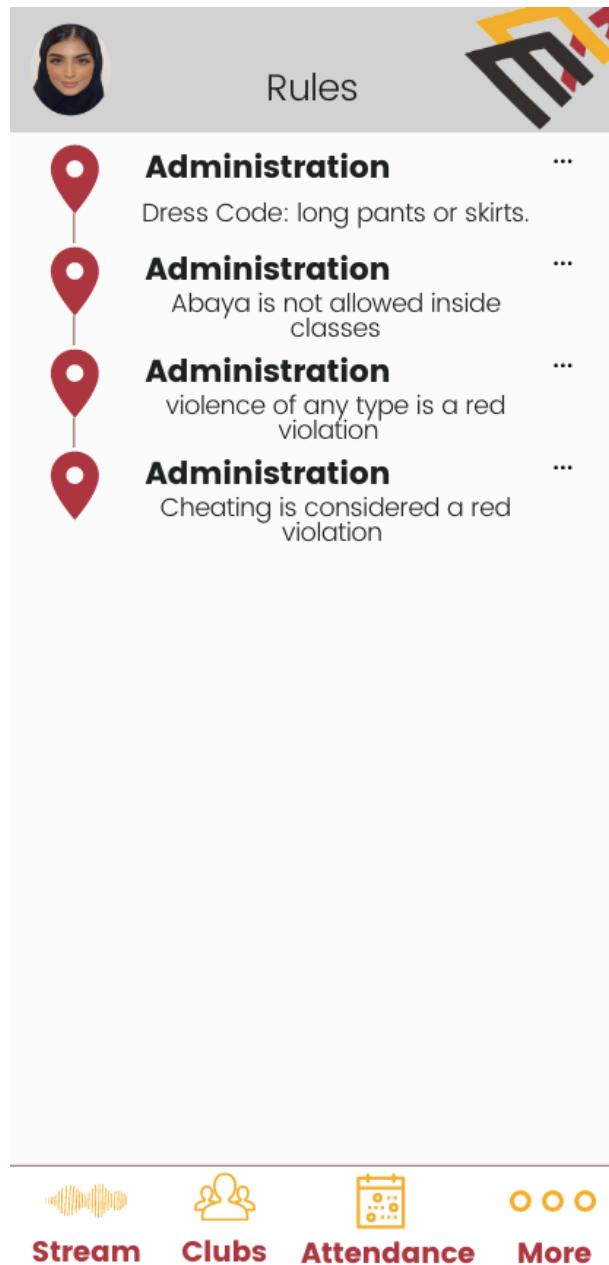


Figure 24: Student rules

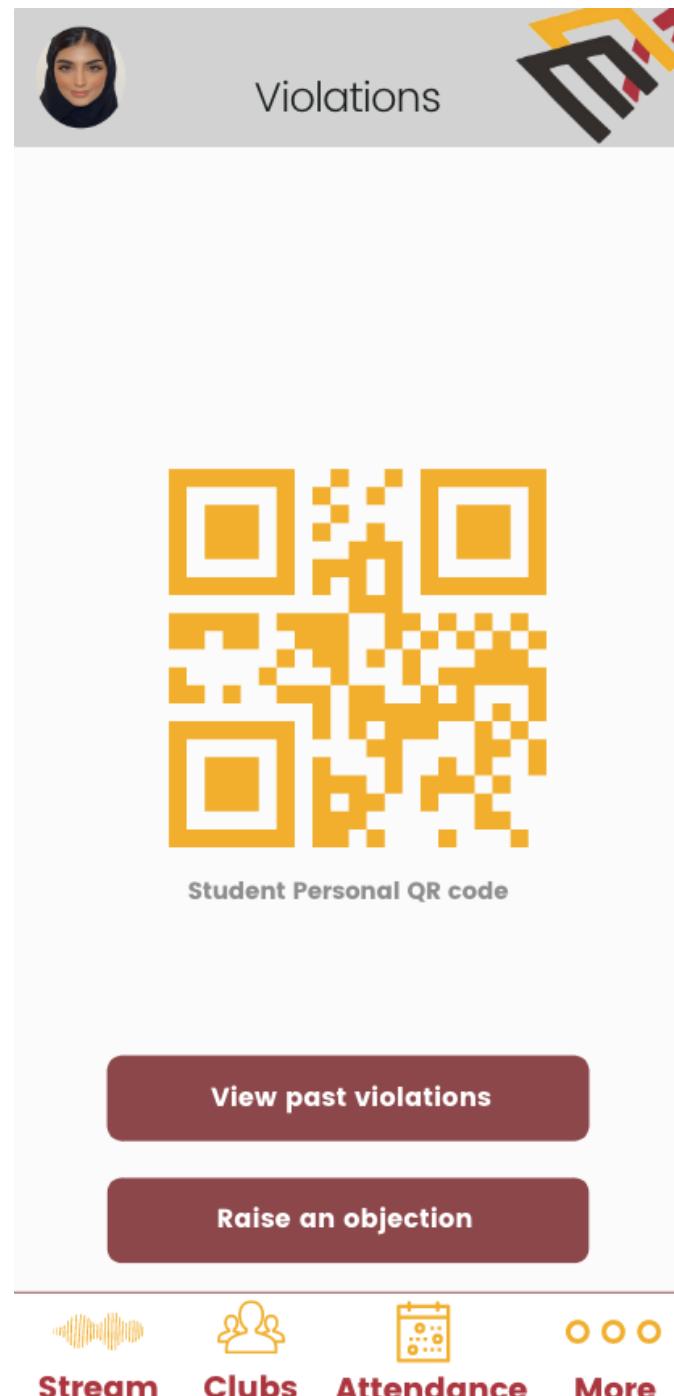


Figure 25: Student violations

#### 4.4 Screen Objects and Actions

This section presents the objects used in CSHJ, their types and their actions as shown below:

#NO	<i>Object</i>	<i>Type</i>	<i>Action</i>
<b>Common Interface</b>			
1	Login	Button	Move to desired interface
2	Sign up	Button	Move to face id interface
3	Don't allow	Button	Cancel the face ID feature
4	Ok	Button	Approve the face ID feature
5	Forgot password	Button	Move to reset password page
6	Arrow	Button	Go to the previous page
7	Done	Button	Finish editing profile
<b>Administration Interface</b>			
8	Plus	Button	Open add a post page
9	3 dots	List	Choose between edit, delete or pin the post
10	post	Button	Post the desired notification in the stream area
11	All	List	Choose who to share the notification with
12	Cancel	Button	Cancel the post
13	Plus	Button	Move to add a club page

14	Add	Button	Adding a new student club
15	Dismiss	Button	Dismiss the attendance application
16	View	Button	View the attendance excuse paper to review
17	Plus	Button	Adding a new rule or regulation to the rule streaming page
<b>Faculty members Interface</b>			
18	Plus	Button	Adding a new post to the streaming area
19	Edit	Button	Edit the information of the desired class
20	View	Button	View the details of the desired class
21	Plus	Button	Open add a new class page
22	Add	Button	Adding a new class successfully
23	Check	Check box	Check if the student has attended the class
<b>Security members Interface</b>			
24	Add	Button	Adding a new violation ticket
<b>Students Interface</b>			
25	Plus	Button	Add a new post only to other students
26	Plus (join)	Button	Joining a student club

27	View	Button	View the student personal attendance details for the selected class
28	Upload absence excuse	Button	Upload a document, picture or any kind of absence excuse
29	Three dots	List	The ability to share the notification
30	View past violations	Button	Viewing all past violation history
31	Raise an objection	Button	Upload an objection text or a message to the administration for any past violation

Table 4 Screen objects and actions

#### 4.5 Other Interfaces

This section is not applicable to the project.

## 5. System Architecture

This section provides an overview of the System architecture. **CSHJ** platform has four users: students, faculty members, security managers, and administrators. There are some common standard functionality and interfaces between them, such as a homepage, sign-up, login, password retrieve, and verification step. Also, **CSHJ** has other features that will be discussed later. **Figure 26** shows the system architecture for each user.

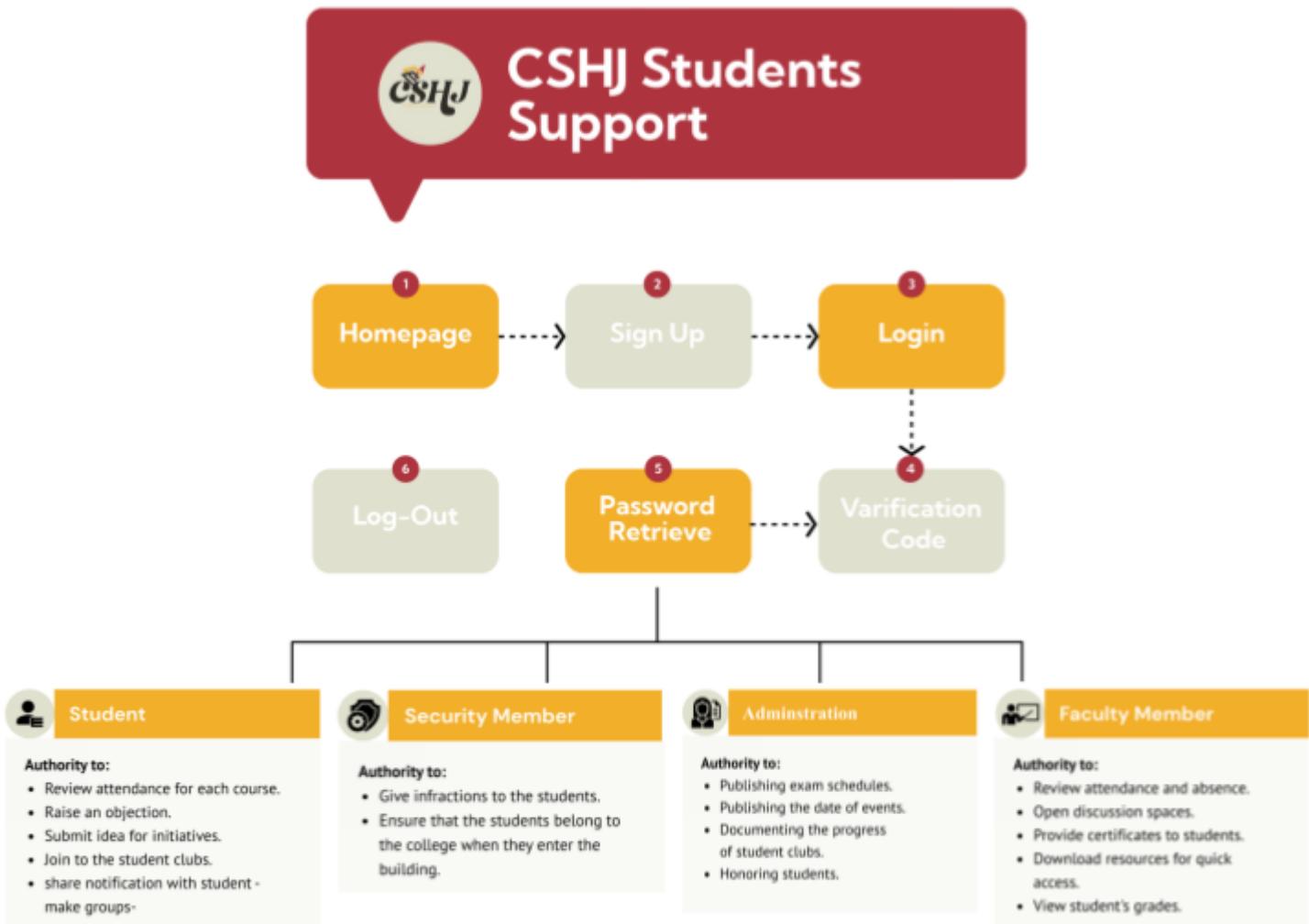


Figure 26: System overview

### 5.1 Architectural Design Approach

The appropriate architectural approach design for the **CSHJ** system is Layered architecture. It is used to simulate the interface between subsystems. Also, it Supports the layer-by-layer gradual construction of subsystems. Only the layer immediately adjacent is impacted when a layer interface changes.

## 5.2 Architectural Design

The architecture has three essential layers that describe the design used in the **CSHJ** system. **Figure 27** shows the relationship between the layers.

**User interface layer:** In this layer, we utilized a GUI to receive user inputs, pass them on to the application services layer, and then show the user the data.

**Application Services layer:** the layer will be bound to the user interface layer and the data layer together, joining the user interfaces notably and backing them up with the data, right before they reach the data layer. At the same time, user interface components primarily in this layer are not obliged to be displayed on screen. The user interface components just need to be aware of the cover they need to offer different kinds of user actions.

**Data Layer:** The data layer is described as having access to the database server and being able to use queries sent from the JVM to the database to save, update, retrieve, and remove data.

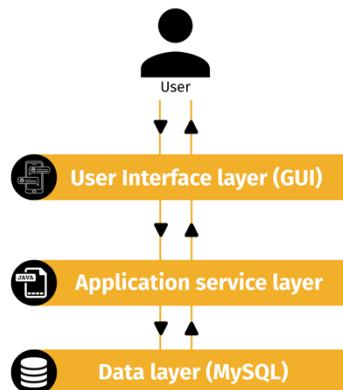


Figure 27: System layers

## 5.3 Subsystem Architecture

### 5.3.1 General view of the system

The figure below shows the general idea of how the two main users will interact with the system.

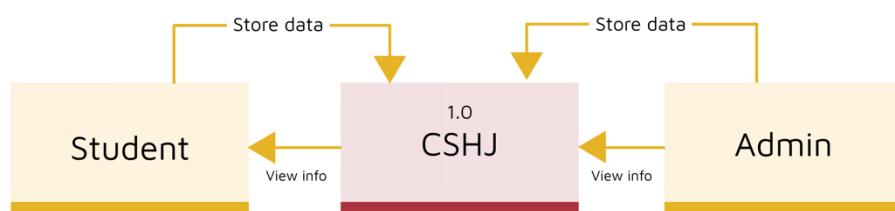


Figure 28: the idea of the interaction with the system

### 5.3.2 Common functionality subsystem

The following diagram describes the functionalities that appear for the users (Student, Administrator, security and faculty members). All the users can login by their email and password. Also, they can set a new password.

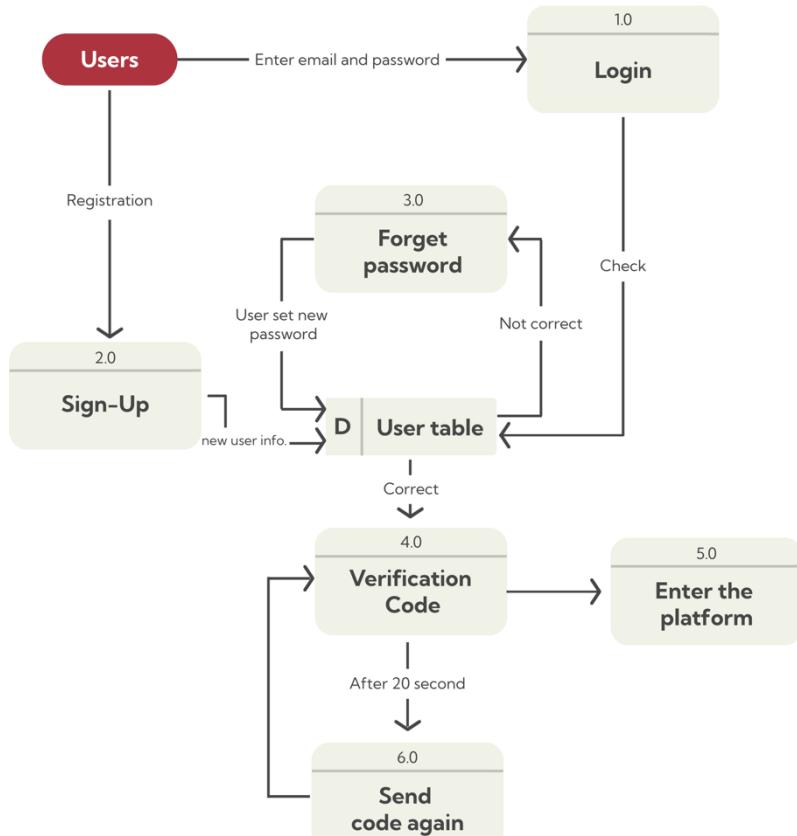


Figure 29 common functionalities

### 5.3.3 Admin subsystem

In figure 30 Admin in the platform has many functions. Admin can view student information, add a grade, add/delete resources, add/update exam schedule, update event information, and give infractions to the students.

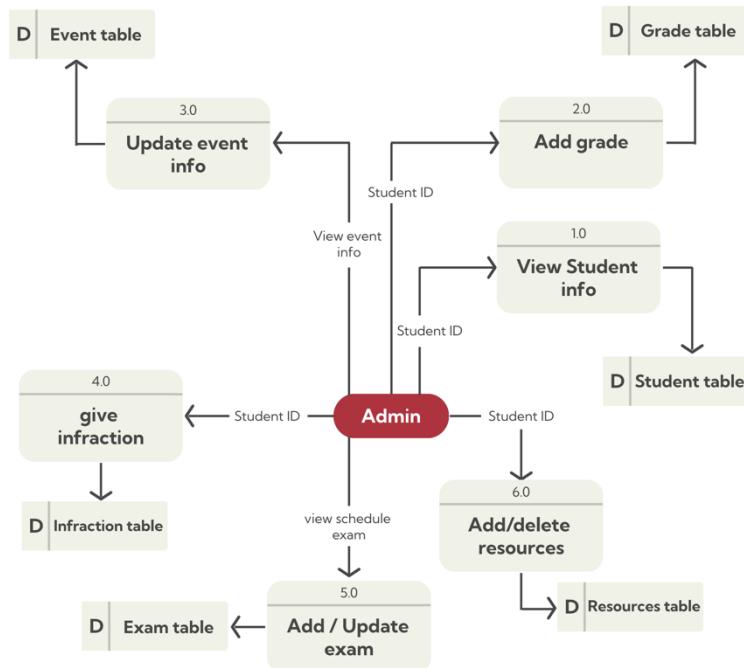


Figure 30 Admin's subsystem

### 5.3.4 Student subsystem

**Figure 31** shows the data flow diagram representing the Student subsystem. Students can view their grades and attendance, join the student clubs in the college, and raise an objection to the violations that were detected.

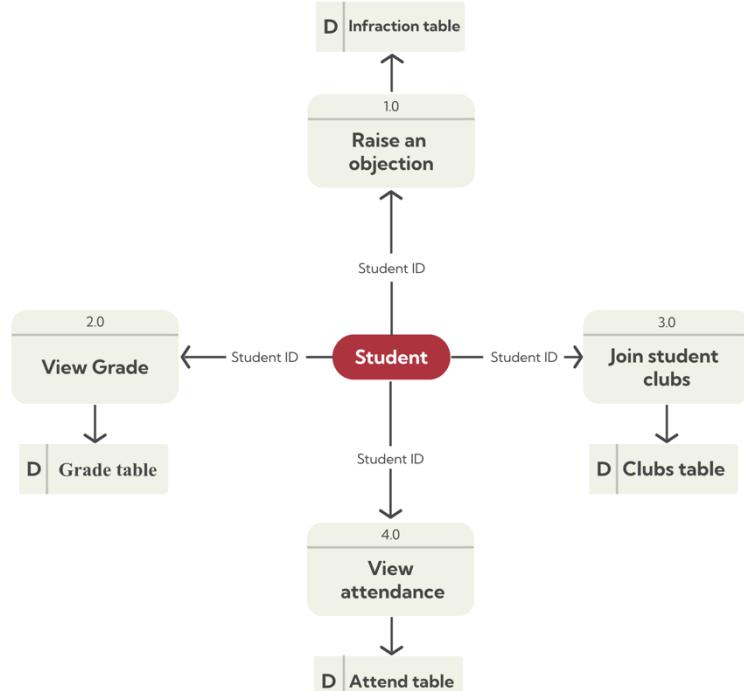


Figure 31 student's subsystem

## 6. Data Design

This section describes the Data Description, Data Dictionary, and Database Description.

### 6.1 Data Description

This section presents the entities, their attributes, their types as well as their constraints by using the SQL as shown in **Table 5** below:

Entity	Field	Type	Primary Key	Foreign Key	Constraints
College	College_ID	INTEGER	✓		Not Null, unique
	College_name	VARCHAR(20)			Not Null
Administration	Admin_ID	INTEGER	✓		Not Null, unique
	Fname	VARCHAR(20)			Not null
	Lname	VARCHAR(20)			Not null
	Email	VARCHAR(40)			Not null
	Password	VARCHAR(20)			Not null
	College_ID	INTEGER		✓	Not Null, unique
	Event_ID	INTEGER		✓	Not Null, unique

<b>Faculty Member</b>	Faculty_ID	INTEGER	✓		Not Null, unique
	Fname	VARCHAR(20)			Not null
	Lname	VARCHAR(20)			Not null
	Email	VARCHAR(40)			Not null
	Password	VARCHAR(20)			Not null
	College_ID	INTEGER		✓	Not Null, unique
	Dep_ID	INTEGER		✓	Not Null, unique
	Course_ID	INTEGER		✓	Not Null, unique
<b>Student</b>	Student_ID	INTEGER	✓		Not Null, unique
	Fname	VARCHAR(20)			Not null
	Lname	VARCHAR(20)			Not null
	Email	VARCHAR(40)			Not null
	Password	VARCHAR(20)			Not null

<b>Infraction</b>	College_ID	INTEGER		✓	Not Null, unique
	Course_ID	INTEGER		✓	Not Null, unique
	Inf_ID	INTEGER	✓		Not Null, unique
	Inf_type	VARCHAR(20)			Not null
	Student_ID	INTEGER		✓	Not Null, unique
	Security_ID	INTEGER		✓	Not Null, unique
	Inf_date	DATE			Not null
<b>Security Members</b>	Security_ID	INTEGER	✓		Not Null, unique
	Fname	VARCHAR(20)			Not null
	Lname	VARCHAR(20)			Not null
	Email	VARCHAR(40)			Not null
	Password	VARCHAR(20)			Not null
	College_ID	INTEGER		✓	Not Null, unique

<b>Club</b>	Club_ID	INTEGER	✓		Not Null, unique
	Club_name	VARCHAR(20)			Not null
	Student_ID	INTEGER		✓	Not Null, unique
<b>Course</b>	Course_ID	INTEGER	✓		Not Null, unique
	Course_name	VARCHAR(20)			Not null
<b>Event</b>	Event_ID	INTEGER	✓		Not Null, unique
	Event_descript ion	VARCHAR(20)			Not null
<b>Department</b>	Dep_ID	INTEGER	✓		Not Null, unique
	Dep_name	VARCHAR(20)			Not null

Table 5 Data Description

## 6.2 Data Dictionary

This subsection describes and presents the data dictionary table of entities, fields, and description of the **CSHJ** database as shown below:

<b>Entity</b>	<b>Field</b>	<b>Description</b>
<b>College</b>	College ID	<b>This field describes the ID of each college. It must be unique</b>
	College name	<b>The college name</b>
<b>Administration</b>	Admin ID	<b>This field describes the ID of the admins which works as the username so they can log in to the system. It must be unique</b>
	First name	<b>The first name of each admin</b>
	Last name	<b>The last name of each admin</b>
	Email	<b>The email of each admin</b>
	Password	<b>The password of each admin so they can log in to the system</b>
	College ID	<b>In order to determine from which college the admin is, the ID of the college must be determined</b>
	Event ID	<b>The ID of each event that the admin will write in the streaming area</b>

<b>Faculty Member</b>	Faculty ID	<b>This field describes the ID of the faculty members which works as the username so they can log in to the system. It must be unique</b>
	First name	<b>The first name of the faculty member</b>
	Last name	<b>The last name of the faculty member</b>
	Email	<b>The email of the faculty member</b>
	Password	<b>The password of the faculty member</b>
	College ID	<b>In order to determine from which college the faculty member is, the ID of the college must be determined</b>
	Department ID	<b>In order to determine from which department the faculty member is, the ID of the department must be determined</b>
	Course ID	<b>In order to determine which courses the faculty member teaches, the ID of the course must be determined</b>
	Student ID	<b>This field describes the ID of the students which works as the username so they can log in to the system. It must be unique</b>
<b>Student</b>	First name	<b>The first name of the student</b>

	Last name	<b>The last name of the student</b>
	Email	<b>The email of the student</b>
	Password	<b>The password of the student</b>
	College ID	<b>In order to determine from which college the student is, the ID of the college must be determined</b>
	Course ID	<b>In order to determine which courses the student has, the ID of the course must be determined</b>
<b>Infraction</b>	Infraction ID	<b>The ID of the infraction</b>
	Infraction type	<b>The type of the infraction</b>
	Student ID	<b>The ID of the student who violates the rules</b>
	Security ID	<b>The ID of the security member that generates the infraction</b>
	Infraction date	<b>The date of the infraction</b>
<b>Security Members</b>	Security ID	<b>This field describes the ID of the security member which works as the username so they can log in to the system. It must be unique</b>
	First name	<b>The first name of the security member</b>

	Last name	<b>The last name of the security member</b>
	Email	<b>The email of the security member</b>
	Password	<b>The password of the security member</b>
	College ID	<b>In order to determine from which college the security member is, the ID of the college must be determined</b>
<b>Club</b>	Club ID	<b>The ID of the club</b>
	Club name	<b>The name of the club</b>
	Student ID	<b>The student ID that enrolled in the club</b>
<b>Course</b>	Course ID	<b>The ID of the course. It must be unique</b>
	Course name	<b>The name of the course</b>
<b>Event</b>	Event ID	<b>The ID of the events. It must be unique</b>
	Event description	<b>The description of the event</b>
<b>Department</b>	Department ID	<b>The ID of the department. It must be unique</b>
	Department name	<b>The department name</b>

Table 6 Data Dictionary

### 6.3 Database Description

This subsection demonstrates the database of the **CSHJ** system by using the Entity-Relationship Diagram (ERD) as shown in **figure 32** and the ER mapping model as shown in **figure 33** that contain the entities, attributes, and the relationship between the attributes [2].

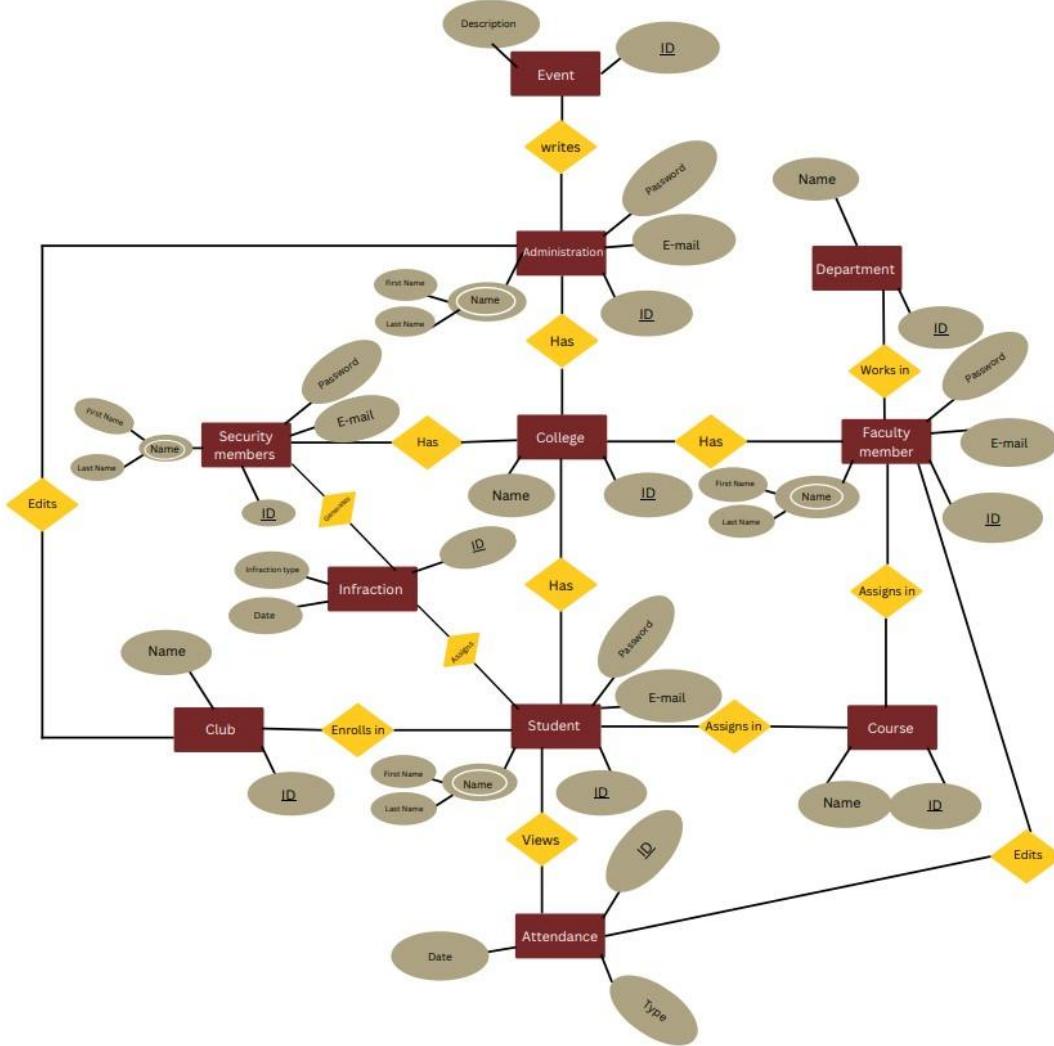


Figure 32 CSHJ ERD

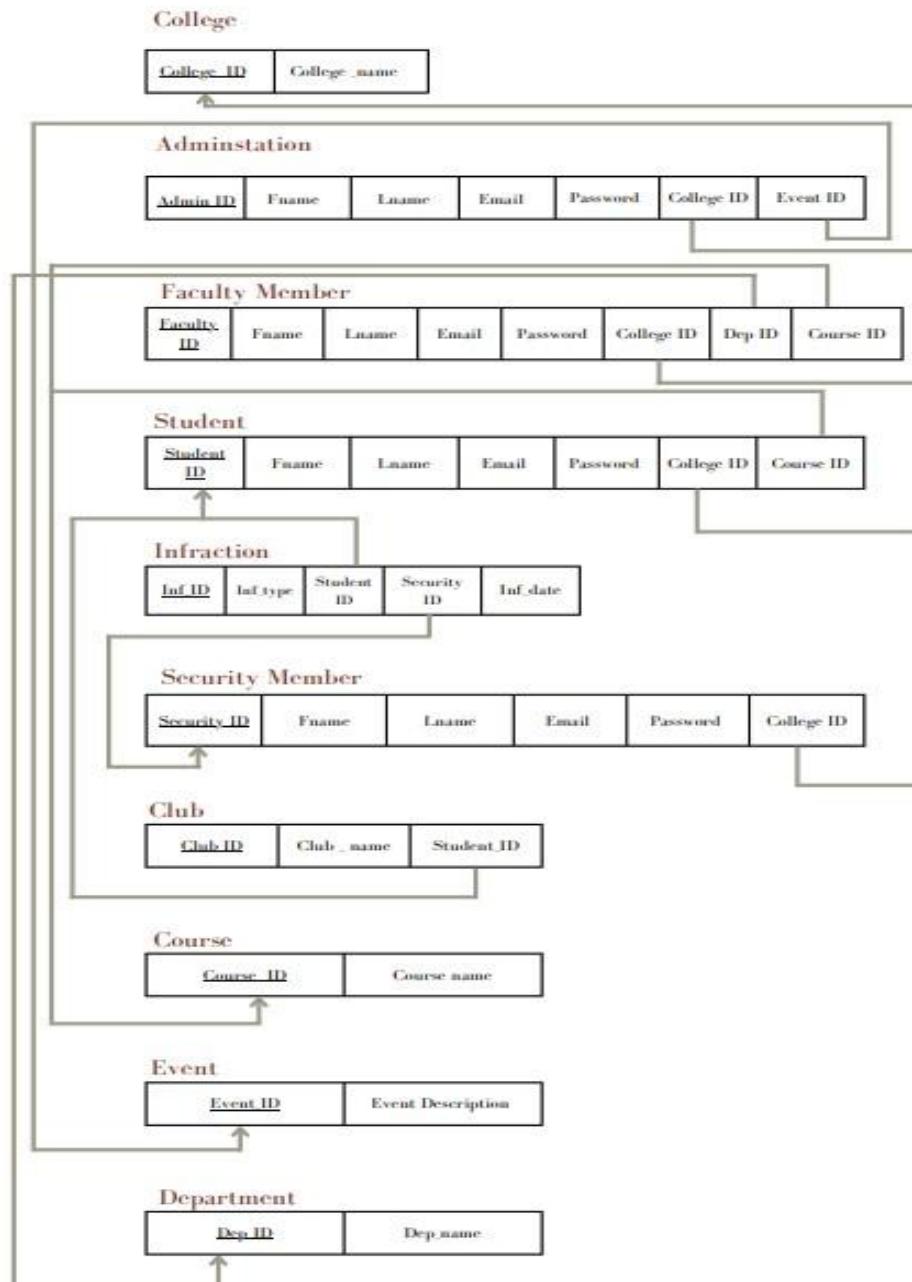


figure 33 ER mapping model

## 7. Component Design

### 7.1 Common functions

#### 7.1.1 Login

*Admin\_login () {*

Enter username and password

If username and password are found in the database Then go to the admin homepage interface

Else

Show error message "Incorrect password or username"

Show error message "please try again!!"

}

*Student\_login() {*

Enter username and password

If username and password are found in the database Then go to the student homepage interface

Else

Show error message "Incorrect password or username"

Show error message "please try again!!"

#### 7.1.2 Sign up

*Sign\_up() {*

Enter username, password, email, position and college name

If the username has already been used show an error message " A username has already been used, try again"

If the password is less than 8, show an error message "password is too short! Enter 8 characters at least"

If email is already used show an error message " an account is already created with this email , try again!"

If next button is pressed and collage name is not entered show an error message " Collage name field is empty!"

Else

Show message " Please verify your email to complete your CSJH account setup." }

### 7.1.3 Forgot password

***forgot\_Password(){***

Enter the email

If the email is found in database

Then send a reset password link to the email Else

Show error message "Email is not found, try again" }

### 7.1.4 Change password

***changePassword () {***

*Enter the old password*

*If the old password found in database*

*Then enter the new password ,confirmed password*

*If (new password == confirm password)*

*Update the old password in database*

*Else*

*Show Error message “The new password doesn’t match the confirmed password!”*

*Else*

*Show Error message “Invalid password!”}*

### 7.1.5 Move to the appropriate interfaces according to the position of the user.

***Proper\_Interface() {***

*If (position == “Student”)*

*Then show student’s interface*

*Else If (position == “Faculty Member”)*

*Then show Faculty Member’s interface*

*Else If (position == “Security Member”)*

*Then show Security Member’s interface*

*Else*

*show Administrator’s interface}*

***editProfile () {***

*If there is invalid input*

*Then show Error message “Invalid input please check the format to this field!”*

*Else Update the fields in database*

*}*

## 7.2 Student functions

#### 7.2.1 Join to club

**join\_Club()** {  
If the club full

Then show the information message " Sorry, you cannot join this club. You can apply to another club!"

Else if student id found in another club

Then show Error message "you cannot join this club, if you want to join you should have no club registration"

Else Show message "Successfully joined the club"

}

#### 7.2.1 Raise an objection

**Raise\_objection()** {

If the student has a violation that was detected by the security members

Then Opening a page for the student to file any objection to the violations

Else

Show Error message " You do not have any violations"

}

#### 7.2.2 Review attendance for each coerce

**Review\_Attendance()** {

If the student id is registered in the course

Then show attendance's page for the course

Else Show Error message " You are not registered in this course"

}

### 7.3 Security member functions

#### 7.3.1 Verify that the student belongs to the college

**Verify ()** {

If the student, whose presence in the college was monitored does not belong to the college

then give infraction to the student

}

### 7.3.2 Give infraction to the students

***Infraction () {***

*If the student commits any violation*

*Then a notification of the violation will be sent to the student's personal account*

***}***

## 7.4 Faculty member functions

### 7.4.1 Review attendance for each coerce

***Review\_Attendance() {***

*If the faculty member id is registered in the course*

*Then show attendance's page for the course*

*Else Show Error message " You are not registered in this course "*

***}***

### 7.4.2 Review grade for students

***Review\_grades() {***

*If the faculty member id and student id are registered in the course*

*Then show grade's page for students*

*Else Show Error message " There is an error. Make sure that the faculty member and student are registered in this course "*

***}***

## 7.5 Administration functions

### 7.5.1 Add exam schedules

***Add\_exam\_schedules(){***

*Enter course name, date, time, and class name*

*If exam schedule is already added show message " exam schedule is already added "*

*Else add exam schedule and show message " exam schedule is added "*

***}***

### 7.5.2 Add event schedules

**Add\_event\_schedules()**{

*Enter event name, date, time, and place*

*If event schedule is already added show message " event schedule is already added"*

*Else add event schedule and show message " event schedule is added"*

}

### 7.5.3 Editing of schedules

**Editing\_of\_schedules()** {

*Enter event or course name*

*If the event or course schedule has already been added, modification or deletion will be allowed*

*Else show message " We didn't find the event or course schedule with that name try again"*

}

## 8. Detailed System Design

In this section we'll describe the components of our system, how they interact, what each does, and where they fit into the overall architecture. We'll also talk about the interfaces between them, the interactions they support, and how they're used.

### 8.1 Classification

This section will contain a list of components and their definitions. Each component will also have a brief description of what they do.

Component	Classification	Definition and responsibility
<b><i>Common Functions</i></b>		
<i>Admin_login</i>	Function	This function will allow the admin to access his account, whether it is faculty members, security members, or college administrators, by entering his ID and password
<i>Student_login</i>	Function	This function will allow the student to access his account either by entering his ID and password
<i>Sign_up</i>	Function	This function allows the user to register on the platform. The user must provide first and last name, password, email, college name, and position. The customer must agree to the terms of use and have read the privacy policy
<i>forgot_Password</i>	Function	This function checks if the user ID entered matches one of the IDs saved in the database
<i>changePassword</i>	Function	This function will allow the user to retrieve the password by email or phone number
<i>Proper_Interface</i>	Function	This function performs an essential operation: the transition to the appropriate interfaces based on the user position specified when the user logs into the system
<i>editProfile</i>	Function	This function allows the user to edit the profile from different sides, add a new

		photo, edit personal information, and more
<b>Student Functions</b>		
<i>join_Club</i>	Function	This function allows the user to join the clubs available in the college. Users can join up to one club if existing members are enough
<i>Raise_objection</i>	Function	This function is available to the student only if he has violations identified by security personnel, where he can submit a request to object to this violation
<i>Review_Attendance</i>	Function	This job is available to the student if he is registered in the course, where he can see his attendance and absence.
<b>Security Member Functions</b>		
<i>Verify</i>	Function	This function allows security members to verify. If the student whose presence in the college was monitored belongs to the college or not
<i>Infraction</i>	Function	This function allows security personnel to raise the violation; if the student commits any violation, then a notification of the violation is sent to the student's account
<b>Faculty Member Functions</b>		
<i>Review_Attendance</i>	Function	This function is available to the faculty member, provided that the faculty member ID is registered in the course so that the course attendance page is displayed
<i>Review_grades</i>	Function	This job is available to a faculty member, provided that the faculty member's ID and student ID card are registered in the course so the grade page is displayed for students
<b>Administration Functions</b>		

<i>Add_exam_schedules</i>	Function	This function allows administrators to enter the course name, date and time, and class name to view the exam schedules for students
<i>Add_event_schedules()</i>	Function	This function allows administrators to enter the event name, date, time, and location to view the event schedules for students
<i>Editing_of_schedules()</i>	Function	This function allows you to modify the schedules of events or exams by entering the event's name or course. If the event or course schedule has already been added, modification or deletion will be allowed

Table 7 Classification

## 8.2 Definition

The specific purpose and semantic meaning of the component. This may need to refer back to the requirements specification.

## 8.3 Responsibilities

The primary responsibilities and/or behavior of this component. What does this component accomplish? What roles does it play? What kinds of services does it provide to its clients? For some components, this may need to refer back to the requirements specification.

## 8.4 Constraints and Composition

This section includes the pre-conditions and post-conditions of each component in the table below.

Component	Constrains	Pre-condition	Post-condition
<b>Common functions</b>			
<i>Admin_login</i>	The user must create an account	Enter the User's Id, Password, email, and position	check if the information is valid or not
<i>Student_login</i>	The user must create an account	Enter the User's Id, Password, email, and position	check if the information is valid or not

<b><i>Sign_up</i></b>	All fields must be filled. - Passwords must be eight characters long, including uppercase and lowercase letters and numbers. - User's id must be unique. - email should follow the proper format.	Enter the user information	The data will be stored in the database of the user
<b><i>forgot_Password</i></b>	The user must create an account	Enter email	Send the code to the email and reset the new password
<b><i>changePassword</i></b>	The password must be eight characters long, including at least: <ul style="list-style-type: none"><li>● 1 Uppercase letter</li><li>● 1 Lowercase letter</li><li>● Numbers</li></ul>	Enter the new password	The new password will be stored in the database of the user
<b><i>Proper_Interface</i></b>	The user must choose his position	Position selection	The user moves according to his position to the appropriate destinations assigned to him
<b><i>editProfile</i></b>	The data must be valid, and all the fields must be filled	Enter personal information	The data will be stored in the database of the user

### Student's Subsystem

<b><i>join_Club</i></b>	That the user is not registered in another club and that there are vacancies to join the club	Log in to the club by entering personal information and attaching study schedules	A message confirming joining the club will be sent with the information saved in the database
<b><i>Raise_objection</i></b>	That there is already a violation that has been detected by security members	Specify the violation's type, date, and name of the security member, and	The objection request will be answered within one to three

		attach the objection to this violation for processing	days by sending an email
<b>Review_Attendance</b>	The student must be registered in the course	None	View the student's absence and attendance files

### **Security Member's Subsystem**

<b>Verify</b>	None	None	Monitoring students who are not affiliated with the college
<b>Infraction</b>	None	Write the name of the security member, the date the violation was discovered for the student, and the type of violation that will be displayed in the student's account.	Receipt of the violation notice in the student's personal account

### **Faculty Member's Subsystem**

<b>Review_Attendance</b>	The faculty member and the student must be registered in the course	Enter the subject name or code to view attendance and absence	Displays the attendance and absence files of the faculty member
<b>Review_grades</b>	The faculty member and the student must be registered in the course	Enter the subject name or code to view the student's grade	Displays the course grades for the faculty member and the student

### **Administration's Subsystem**

<b>Add_exam_schedules</b>	None	Enter the course name, date and time, and class name	view the exam schedules for students
---------------------------	------	--	--------------------------------------

<i>Add_event_schedules()</i>	None	Enter the event name, date, time, and location	view the event schedules for students
<i>Editing_of_schedules()</i>	Course schedule has already been added	Modify the schedules of events or exams by entering the event's name or course.	View the event/exam schedules for students

**Table 8** Constraints and Composition

## 8.5 Uses/Interactions

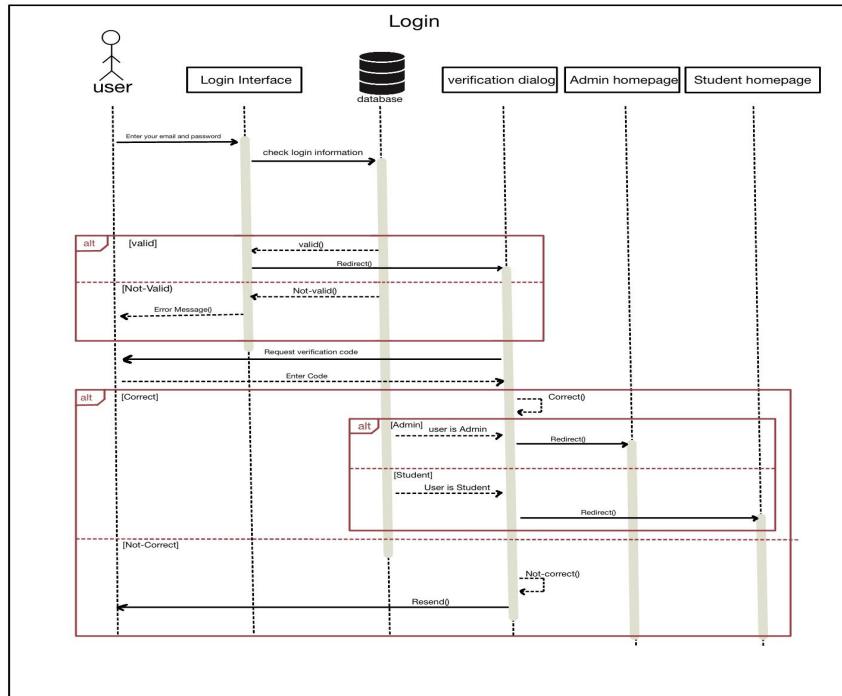
A description of these components collaborations with other components. What other components is this entity used by? What other components does this entity use (this would include any side-effects this entity might have on other parts of the system)? This concerns the method of interaction as well as the interaction itself. Object-oriented designs should include a description of any known or anticipated subclasses, super classes, and meta classes.

### 8.5.1 Common Functions

This subsection presents the common functionalities between admin and student.

### 8.5.1.1 Login Functionality

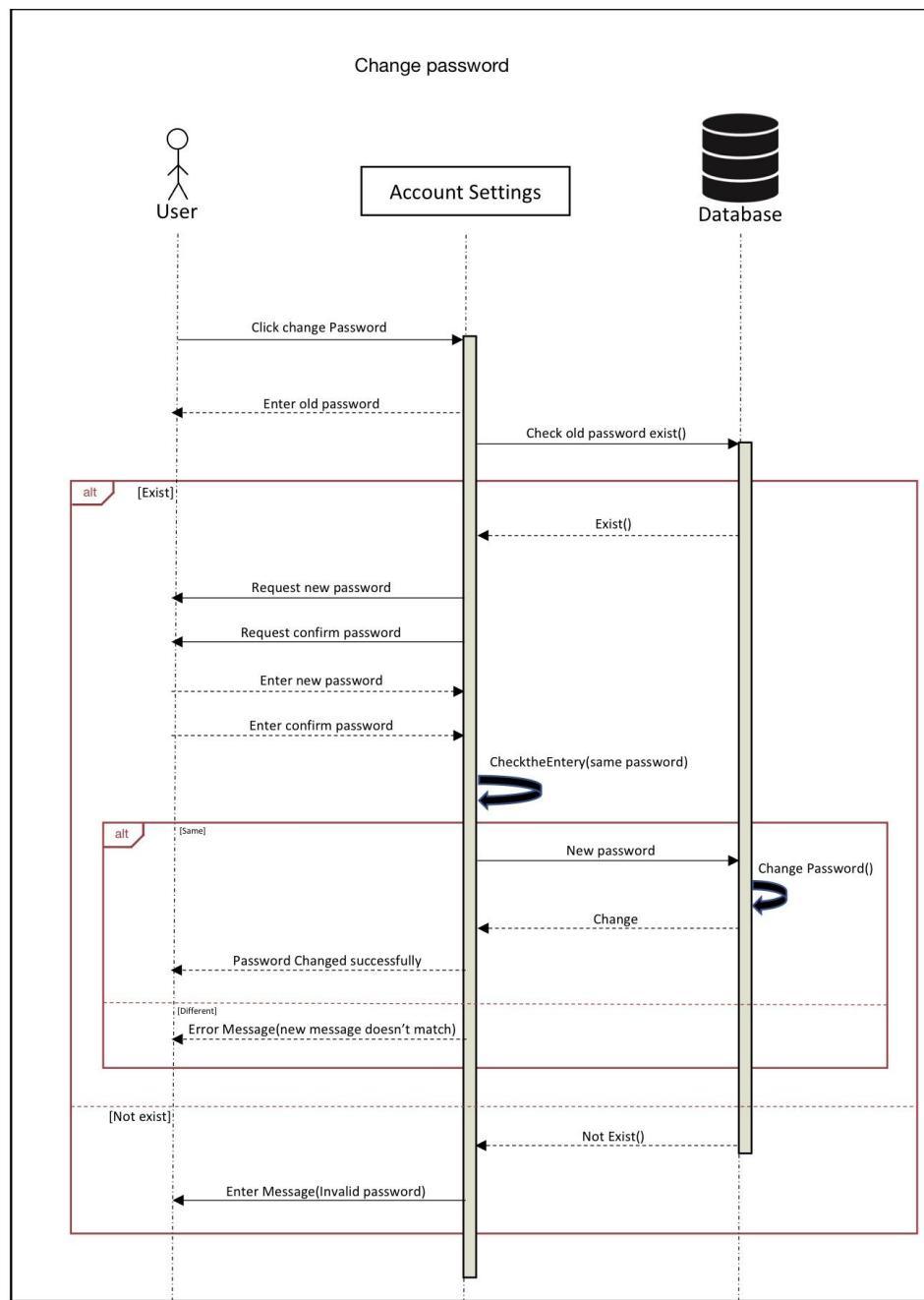
**Figure 34** shows the sequence diagram for login functionality.



**Figure 34** login functionality

### 8.5.1.2 Change Password Functionality

**Figure 35** shows the sequence diagram for changing password functionality.



**Figure 35** changing password functionality.

### 8.5.1.3 Edit Profile Functionality

**Figure 36** shows the sequence diagram for edit profile functionality.

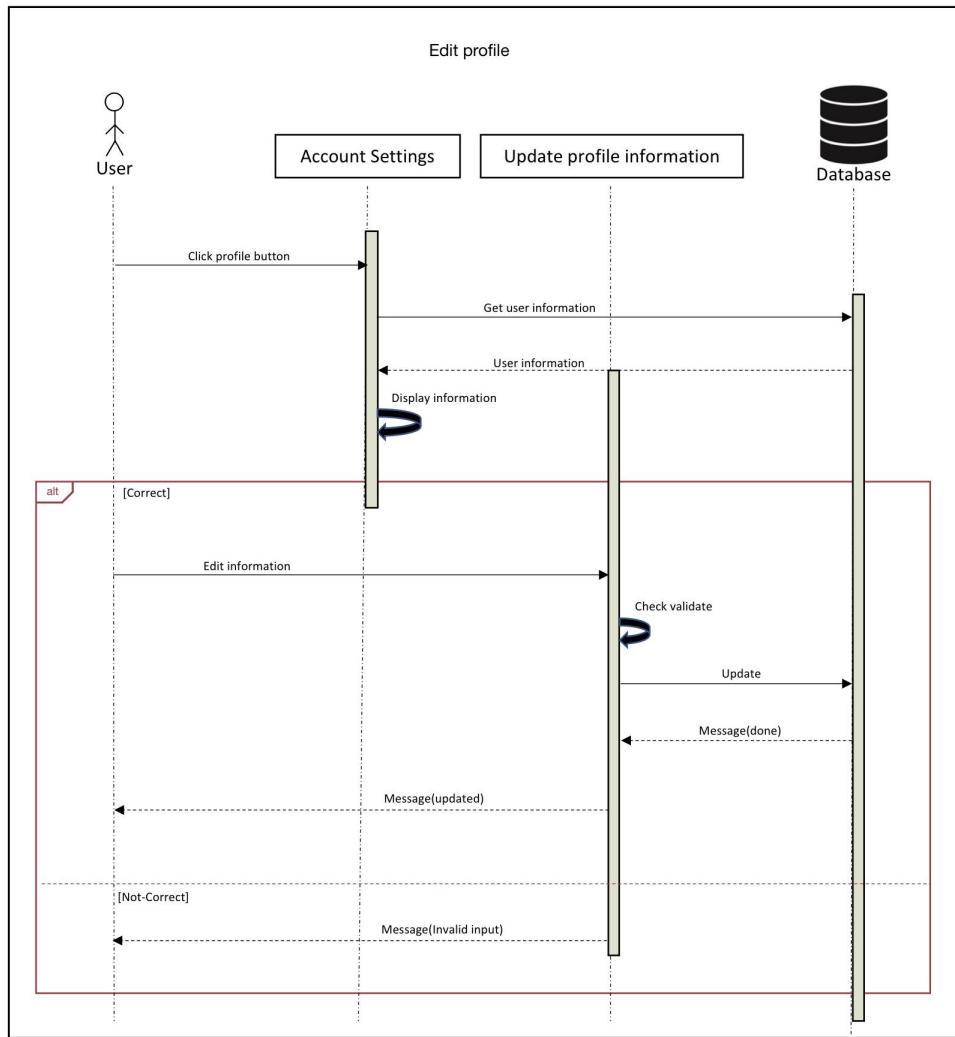


Figure 36 edit profile functionality

### 8.5.2 Admin's Functions

This subsection presents the functionalities for the admin.

#### 8.5.2.1 Add new grade

**Figure 37** shows the sequence diagram of adding a grade to the student.

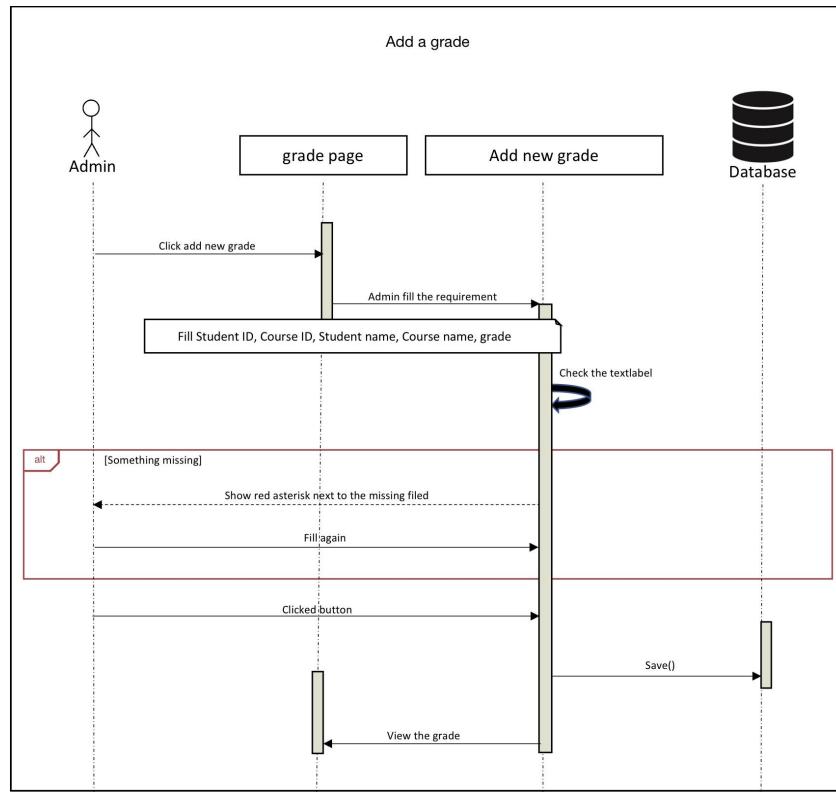


Figure 37 adding a grade to the student

### 8.5.2.2 Create and update event information

**Figure 38** shows the sequence diagram of create and update event information.

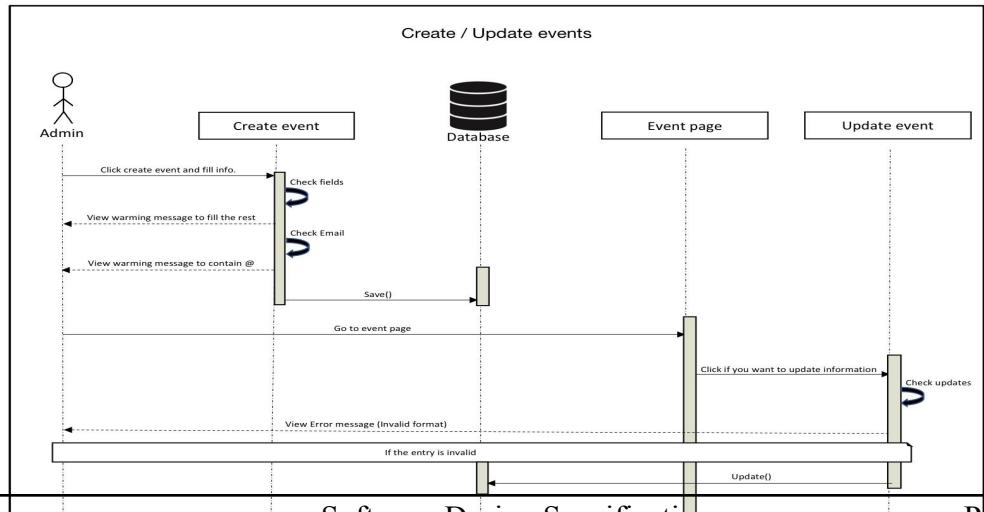


Figure 38 create and update event information

### 8.5.2.3 Add Infraction

**Figure 39** shows the sequence diagram of giving infraction to the student.

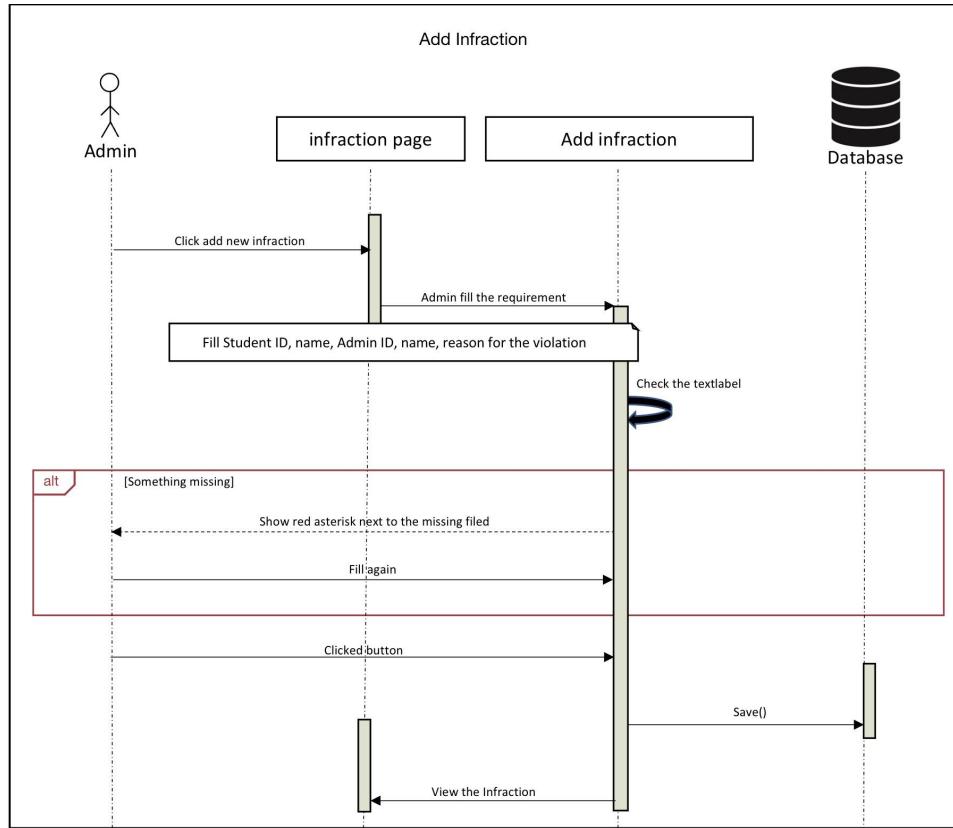


Figure 39 giving infraction to the student

### 8.5.2.4 Add the resources

**Figure 40** shows the sequence diagram of adding the resources to the student.

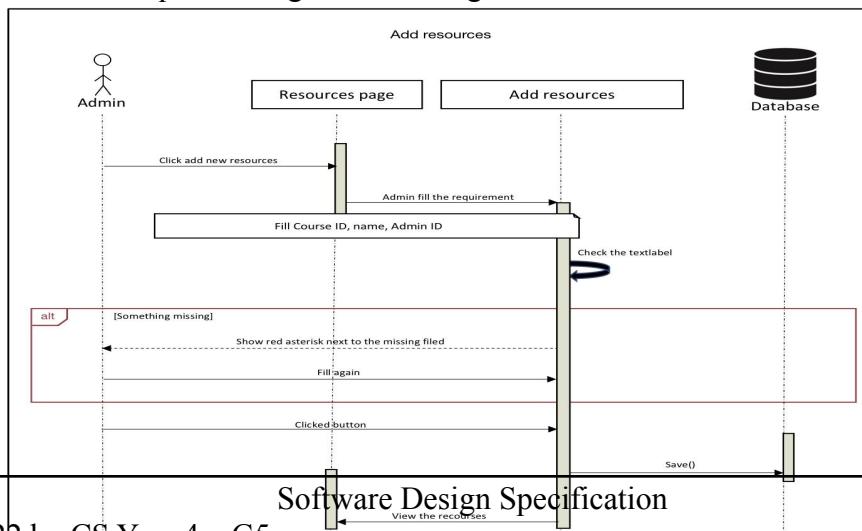


Figure 40 adding the resources to the student

#### 8.5.2.5 Delete a resource

Figure 41 shows the sequence diagram of deleting a resource.

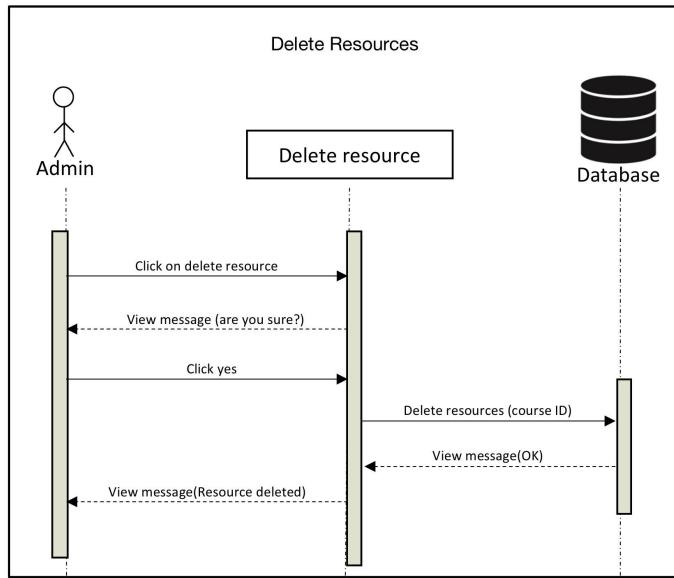


Figure 41 deleting a resource

#### 8.5.2.6 Add exams

Figure 42 shows the sequence diagram of adding exams to the student.

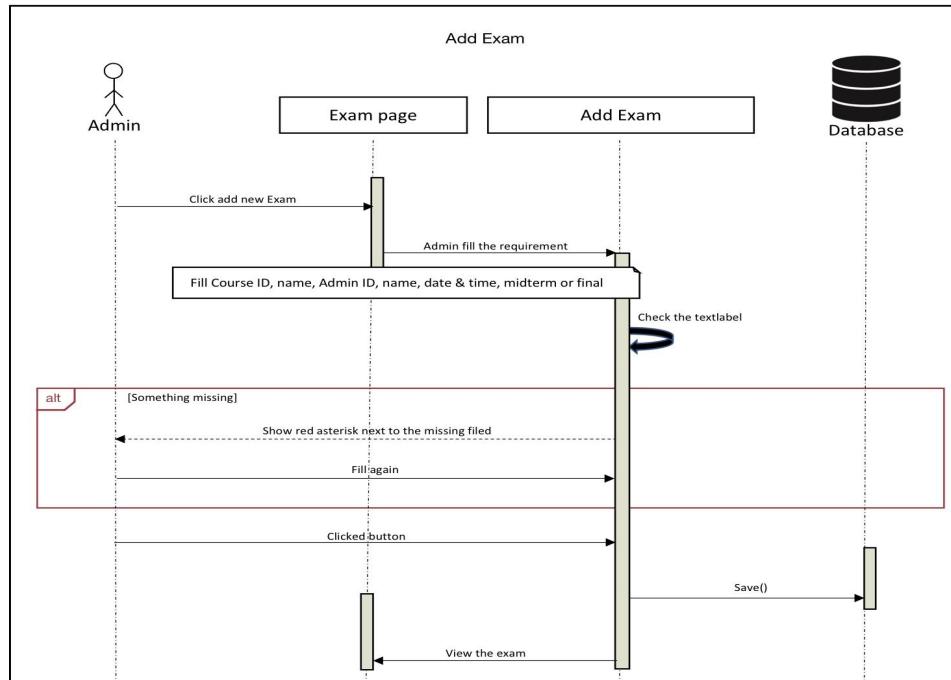


Figure 42 exams to the student.

### 8.5.3 Student's Functions

This subsection presents the functionalities for the admin.

#### 8.5.3.1 Raise an objection

**Figure 43** shows the sequence diagram to raise an objection to the violations that were detected.

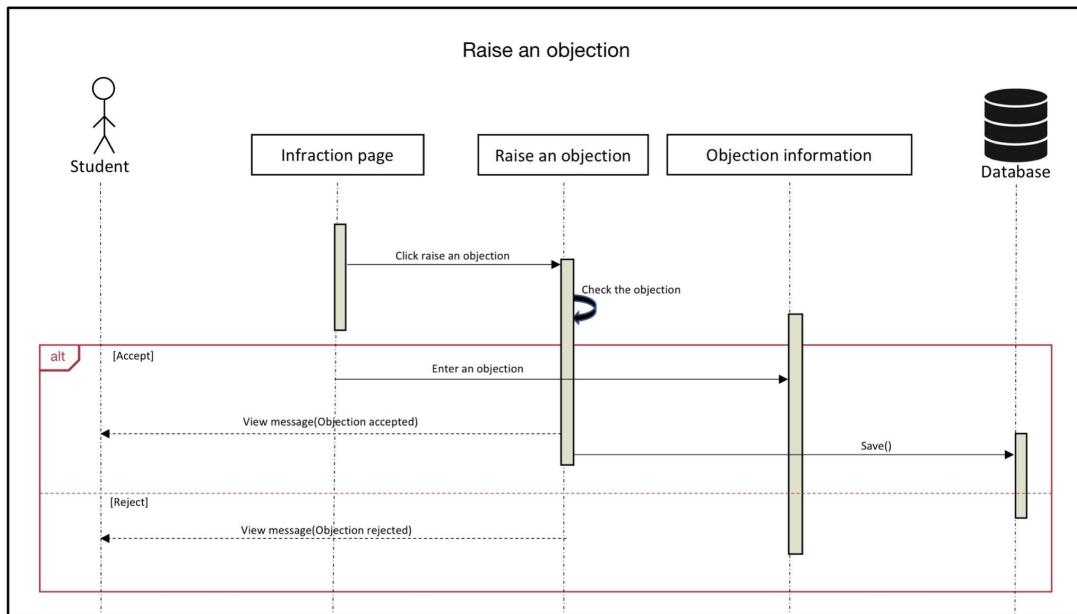


Figure 43 raise an objection

#### 8.5.3.2 View grade

**Figure 44** shows the sequence diagram of viewing grade in a specific subject.

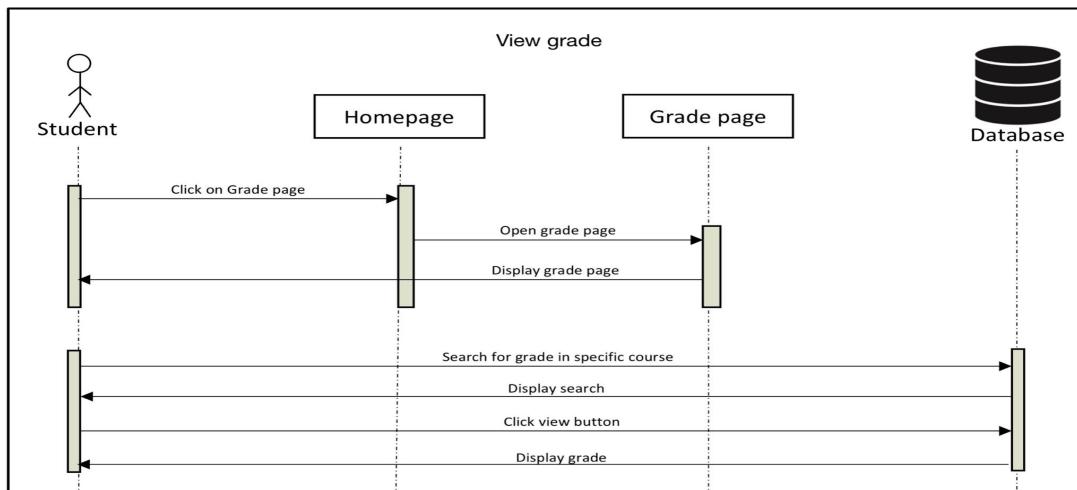


Figure 44 viewing grades

### 8.5.3.3 View attendance

**Figure 45** shows the sequence diagram of viewing attendance in a specific subject.

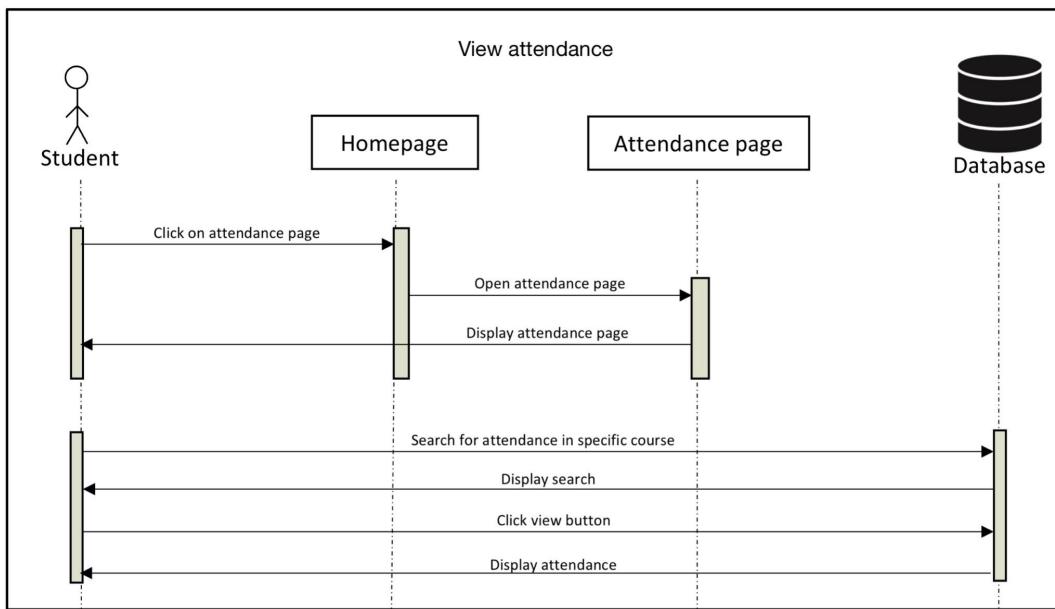


Figure 45 viewing attendance

### 8.5.3.4 Join to student clubs

**Figure 46** shows the sequence diagram of Joining to student clubs.

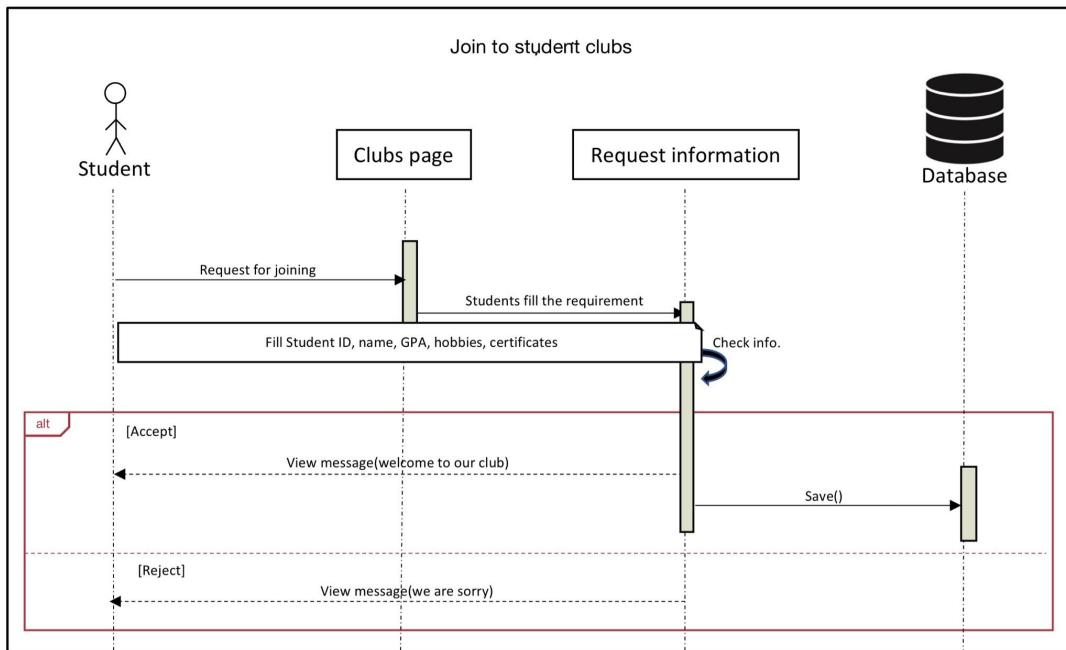


Figure 46 Joining to clubs

## 8.6 Resources

This section will present an overview of the resources used by the CSHJ AI approach-based system. Table 3 shows the resource specifications that are managed, affected or needed by this system.

Type	Resource
Database	MySQL
Operating System	IOS, windows, mac, and android
Memory Storage	1.9 GB
Server	XAMPP
Internet connection	Network

Table 9 Resources

Table 10 shows the race conditions and deadlock situations that may occur and how could be resolved.

Rase conditions and Deadlock	Description	Solution
Network connection	When the device doesn't connect to the internet	Notify the user to check the internet connection and try again
Access data	When more than one student registers for the last vacant seat in one of the clubs announced in the application	The student who completes the registration confirmation process sent to the email first will get the seat

Table 10 Rase conditions and Deadlock

## 8.7 Processing

This subsection describes the **CSHJ** platform components and their descriptions, inputs, and outputs.

### 8.7.1 Common Functions

#### 8.7.1.1 Admin\_login

Table 11 below shows the description, input, output, and limitations of the Login component.

Description	Faculty members, security members, and college administrators can log in to the application.
Input	Click on the login button and decide to inter with Email. In addition to entering verification code.

<b>Output</b>	The home page will be displayed based on the position type
<b>Constraint</b>	If the user entered incorrect data. Such as, Email, Password, Vitrification code.

Table 11 Login component

#### 8.7.1.2 Student\_login

**Table 12** below shows the description, input, output, and limitations of the Login component.

<b>Description</b>	Students can log in to the application.
<b>Input</b>	Click on the login button and decide to inter with Email. In addition to entering verification code.
<b>Output</b>	Student home page will be displayed
<b>Constraint</b>	If the user entered incorrect data. Such as, Email, Password, Vitrification code.

Table 12 Login component

#### 8.7.1.3 Sign\_up

**Table 13** below shows the description, input, output, and limitations of the sign up component.

<b>Description</b>	Customer can create a new account
<b>Input</b>	First name, last name, password, email, phone number, position, and college name. The user must agree to the terms of use and have read the privacy policy.
<b>Output</b>	user will be added to system
<b>Constraint</b>	If some fields are missing, a red asterisk will be shown.

Table 13 sign up component

#### 8.7.1.4 Change\_Password

**Table 14** below shows the description, input, output, and limitations of the change password component.

<b>Description</b>	The user will be able to retrieve his password only if the email matches the data in our database
<b>Input</b>	Email
<b>Output</b>	An email be received contains the new password link
<b>Constraint</b>	If the email is not in the database, the error message will be displayed "Invalid email , try again."

Table 14 change password component

#### 8.7.1.5 Forget\_Password

**Table 15** below shows the description, input, output, and limitations of the forget password component.

<b>Description</b>	Admin and Student can change their password
<b>Input</b>	Enter old, new passwords and confirmation passwords.
<b>Output</b>	Change password
<b>Constraint</b>	If the user entered an incorrect old password or confirmation password.

Table 15 forget password component

### 8.7.1.6 Proper\_Interface

**Table 16** below shows the description, input, output, and limitations of the Proper Interface component.

<b>Description</b>	Transition to the appropriate interfaces based on the user position specified when the user logs into the system
<b>Input</b>	User position
<b>Output</b>	User's home page
<b>Constraint</b>	The user must choose the right position

**Table 16** Proper Interface component

### 8.7.1.7 Edit Profile

**Table 17** below shows the description, input, output, and limitations of the Edit Profile component.

<b>Description</b>	Admin and student can edit profile information
<b>Input</b>	Click on account setting
<b>Output</b>	The modified user information is saved in the database
<b>Constraint</b>	The error message "Data is invalid, try again" will be displayed if the user enters incorrect data.

**Table 17** Edit Profile component

## 8.7.2 Student Functions

### 8.7.2.1 join\_Club

**Table 18** shows the description, input, output, and constraint of the join club component.

<b>Description</b>	Students can join college clubs
<b>Input</b>	Click on Join button
<b>Output</b>	The student will be added to the club and a confirmation message will be sent via email
<b>Constraint</b>	A student cannot join more than one club

Table 18 join club component

### 8.7.2.2 Raise\_objection

**Table 19** shows the description, input, output, and constraint of the raise objection component.

<b>Description</b>	The student can file an objection to a violation that was detected by the security members of the college
<b>Input</b>	Click on the objection button, then specify the type and date of the violation and the name of the security member
<b>Output</b>	The objection request will be raised through the system for processing and response
<b>Constraint</b>	-

Table 19 raise objection component

### 8.7.2.3 Review\_Attendance

**Table 20** shows the description, input, output, and constraint of the review attendance component.

<b>Description</b>	Students can see the attendance and absence of the courses they study
<b>Input</b>	Clicking the button to see my attendance and absence
<b>Output</b>	Attendance and absence files
<b>Constraint</b>	The student must be registered in the course

Table 20 review attendance component

### 8.7.3 Security Member Functions

#### 8.7.3.1 Verify/Infraction

**Table 21** shows the description, input, output, and constraint of the Verify/Infraction component.

<b>Description</b>	Security member can check that students belong to the respective college as well as give them violations
<b>Input</b>	Write the name of the security member, the date the violation was discovered for the student, and the type of violation that will be displayed in the student's account
<b>Output</b>	Warning of a new violation on the student's personal account
<b>Constraint</b>	-

Table 21 Verify/Infraction component

### 8.7.4 Faculty member Functions

#### 8.7.4.1 Review\_Attendance/grade

**Table 22** shows the description, input, output, and constraint of the review attendance/grade component.

<b>Description</b>	Faculty members can review attendance and absence and view students' grades in their accounts in the application
<b>Input</b>	Enter the course code or name
<b>Output</b>	View grades and attendance for students
<b>Constraint</b>	The faculty member and the student must be enrolled in the course

Table 22 review attendance/grade component

### 8.7.5 Administration Functions

#### 8.7.5.1 Add Exam/Event Schedules

**Table 23** shows the description, input, output, and constraint of the add exam/event schedules component.

<b>Description</b>	College administrators can add and manage schedules for events and exams
<b>Input</b>	Enter the course name, date and time, and class name (Exam) Enter the event name, date, time, and location (Event)
<b>Output</b>	View the event/exam schedules for students

<b>Constraint</b>	-
-------------------	---

**Table 23** add exam/event schedules component

## 8.8 Interface/Exports

The subsystem functionality of "CSHJ" must be known to the system to ensure that the system works as designed. In addition, each system component's classification, definition, responsibilities, limitations, and composition are explained clearly and clearly. Section 8.4 discusses constraints, preconditions, and postconditions. Section 8.7 covers the CSHJ input, output, and description specifications. Here we show the activity diagram associated with our application.

### 8.8.1 Sign\_up interface:

<b>Actors</b>	-Students  -Admin (Faculty member, Security member, Administrators)
<b>Description</b>	The user must provide first and last name, password, email, college name, and position. The customer must agree to the terms of use and have read the privacy policy
<b>Data</b>	- Name - Email - Password - Collage Name - Position
<b>Stimulate</b>	Click on sign up button
<b>Response</b>	Admin: accessing the admin homepage, and then they can use the privileges provided to the admin.

	Students: access the student homepage and use the functionalities that are available to them.
<b>Comment</b>	In case of invalid data, an error dialog box will be displayed with the message “Incorrect data, please try again”

Table 24 Sign up interface component

### 8.8.2 Login Interface:

<b>Actors</b>	<ul style="list-style-type: none"> <li>- Students</li> <li>- Admin (Faculty member, Security member, Administrators)</li> </ul>
<b>Description</b>	The user must provide Email/Id and Password
<b>Data</b>	<ul style="list-style-type: none"> <li>- Email/id</li> <li>- Password</li> </ul>
<b>Stimulate</b>	Click on log in button
<b>Response</b>	<p>Admin: accessing the admin homepage, and then they can use the privileges provided to the admin.</p> <p>Students: access the student homepage and use the functionalities that are available to them.</p>
<b>Comment</b>	In case of invalid data, an error dialog box will be displayed with the message “Incorrect data, please try again”

Table 25 Login interface component

### 8.8.3 Forget password Interface:

<b>Actors</b>	<ul style="list-style-type: none"> <li>- Students</li> <li>- Admin (Faculty member, Security member, Administrators)</li> </ul>
<b>Description</b>	The user will be able to retrieve their password only if the email entered matches the data in the database
<b>Data</b>	<ul style="list-style-type: none"> <li>- Enter old</li> <li>- New passwords</li> <li>- Confirmation passwords</li> </ul>
<b>Stimulate</b>	Click on forget password button
<b>Response</b>	An email or SMS message will be sent to the user with the new password
<b>Comment</b>	If the entered email was invalid, then an error message will be displayed “invalid email”

Table 26 Forget password component

#### 8.8.4 Face ID interface:

<b>Actors</b>	<ul style="list-style-type: none"> <li>- Students</li> <li>- Admin (Faculty member, Security member, Administrators)</li> </ul>
<b>Description</b>	This page allows the user to activate the facial recognition feature to facilitate the login process
<b>Data</b>	Open the camera to recognize the user's face

<b>Stimulate</b>	Click on forget ok button
<b>Response</b>	The user will be logged in automatically
<b>Comment</b>	-

**Table 27** Face ID component**8.8.5 join\_Club interface:**

<b>Actors</b>	- Students
<b>Description</b>	The student is allowed to join the clubs available in the college
<b>Data</b>	Entering personal information and attaching study schedules
<b>Stimulate</b>	Click on join button
<b>Response</b>	The club will be added to the club interface in the student's personal account
<b>Comment</b>	-

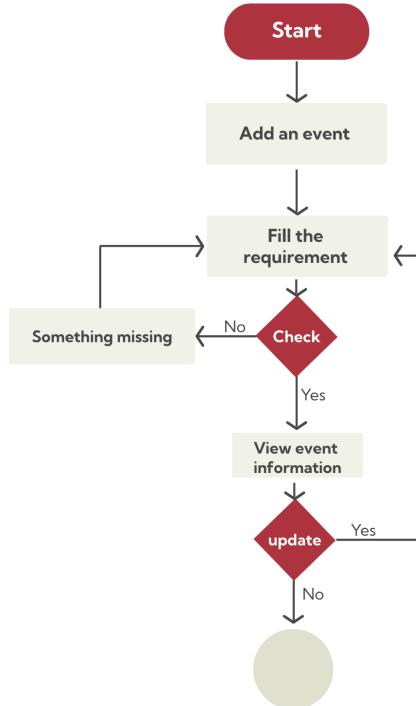
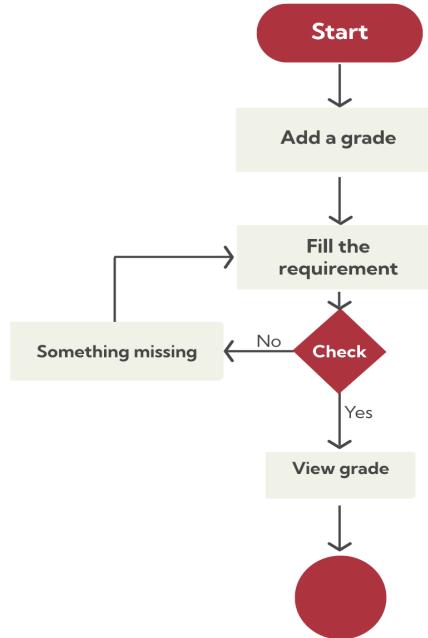
**Table 28** join club interface**8.8.6 Edit Profile interface:**

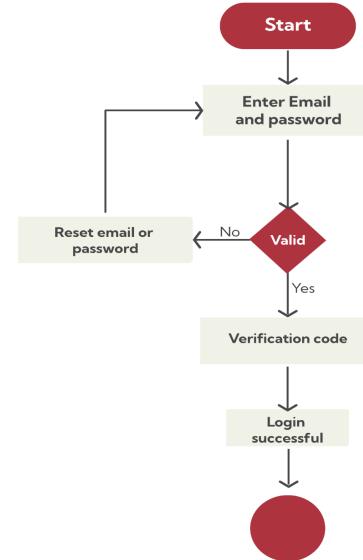
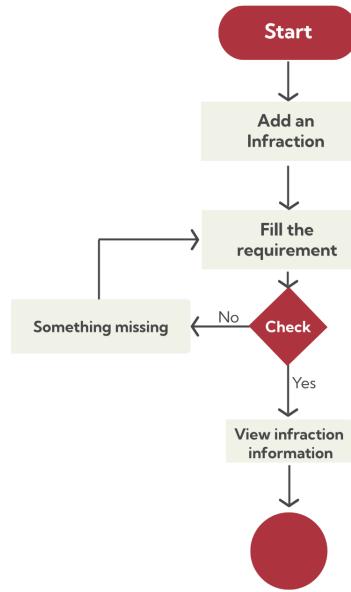
<b>Actors</b>	- Students - Admin (Faculty member, Security member, Administrators)
<b>Description</b>	This function allows the user to edit the profile from different sides, add a new photo, edit personal information, and more
<b>Data</b>	Entering personal information(name, email, collage name, position)
<b>Stimulate</b>	Click on done button
<b>Response</b>	The user's personal data will be modified
<b>Comment</b>	-

Table 29 edit profile

## 8.9 Detailed Subsystem Design

This section describes the main components and data flow of the platform, the following flowchart shows the admin and student subsystems.





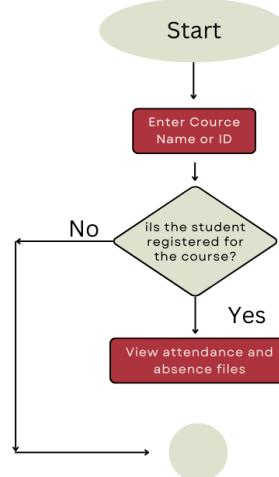
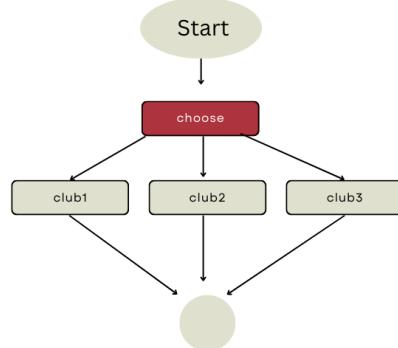
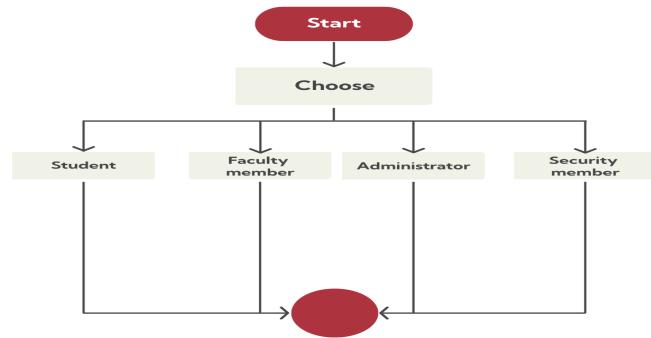


Figure 47 Flow charts of the system

## 9. Other Design Features

Earlier, we did not cover the architectures of PIPs and filters that describe how functional transformations process inputs to produce outputs. The figure below shows the certification pip and filter architecture:

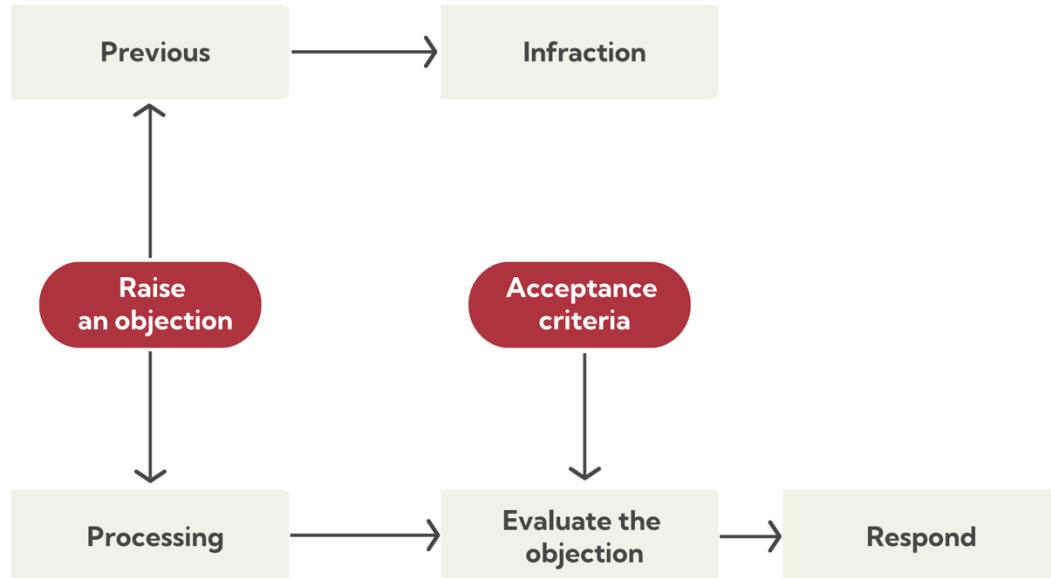


Figure 48 the certification pip and filter architecture

## 10. Requirements Traceability Matrix

<b>Associated ID in SRS</b>	<b>Observed in</b>	<b>Technical Assumption(s) and/or Customer Need(s)</b>	<b>Functional requirement</b>	<b>System component</b>
3.2	Common	The user can log in by providing a valid username and password.	Log-in	Login
		The user can register by entering the required personal information on the sign-up page	Sign-up	Sign up
		If a user registers, they may then use their phone number or an academic email to get their password reset.	Password Retrieve	Password Retrieve
		It is safe to change the password every six months. The user can change the password whenever wants.	Password change	Password change
		The user can edit their profile such as modifying the phone number.	Edit Profile	Profile
3.2.5	User Student 1:	Students can view all the college's clubs, and their descriptions, and	Join to club	Clubs

		sign up for the one they select.		
		A student can submit any objection to the violations submitted to him through the security members.	Raise an objection	Raise an objection
		Students can view and track their attendance for each registered class.	Review Attendance	Attendance
3.2.4	User 2: Security member	The attendance technology will automatically verify student identification.	Verify	Verify
		security members can give the student a violation according to the laws.	Infraction	Infraction
3.2.3	User 3: Faculty member	The faculty member can review attendance at the end of each day, the end of each week, and every month.	Review Attendance	Attendance
		Review and modify the grades of students enrolled in the course.	Review grade for students	Grade's page
3.2.2	User 4: Administration	Adding and modifying the test schedule is done through the administration.	Add exam schedules	Add exam schedules

		Event schedules are announced and added by the administration.	Add event schedules	Add event schedules
		Management could modify schedules.	Editing of schedules	Edit

**Table 30** Requirements Traceability Matrix