SCHOOL OF SCIENCE AND ENGINEERING

ATTENDANT: AUI Health Center Student Records Management System

EGR 4402: CAPSTONE DESIGN

14th December 2022

Asma Dalil

Supervised by Dr Nisar Naeem Sheikh

ATTENDENT

<div align="center">Capstone Report</div>

Student Statement:

I, Asmaa Dalil, have applied ethics to the design process and in the selection of the final proposed design. And that, the designer has held the safety of the public to be paramount and has addressed this in the presented design wherever may be applicable.

<div align="center">Asmaa Dalil</div>

<div align="center">Approved by the Supervisor</div>

<div align="center">Dr. SHEIKH N N</div>

ACKNOWLEDGEMENTS

I want to thank Dr. Sheikh for his tireless efforts and never-ending hard work. Without his direction, this endeavor would not have been possible. Your words of support, counsel, and direction were really helpful in making my capstone project a success. I also want to express my gratitude to Dr. Lamiae Bouanane for being willing to respond to inquiries and offer feedback. Additionally, Dr. Guennoun offered the various papers used in the medical facility. Additionally, I want to thank my parents for going above and beyond by always believing in me and my capacity to succeed despite the ongoing difficulties and obstacles. Last but not least, I would want to sincerely thank my sister Chaima Dalil for her unwavering commitment, love, and honesty.

Table of Contents

List of Figures

ABSTRACT

Al Akhawayn University campus provides students with many facilities to improve their university experience and to engage them in campus activities. The core of the campus and the most significant facility is the Health Center in Building 26. Health Center at Al Akhawayn University has four doctors, four nurses, and an insurance coordinator in the service of Al Akhawayn students. The doctors are on student's service 24 hours a day , 7 days a week. The main mission of the Health Center is to provide healing, caring, and education to the University community through different services such as health and wellness, pharmacy for common medicines, and primary care services.

The main purpose of this project is to ensure the fulfillment of the mission of the Health Center at Al Akhawayn University by providing the doctors and nurses with a web application that digitalizes their job and makes it easier for them to manage the records of students. This application helps students to fill out their medical forms easily and quickly and helps the staff to manage students' records. Attendant will improve the daily services at the Health Center in interaction with students.

1. Introduction

2. Functional requirement specification

In the first few weeks of the semester, I had regular meetings with Dr. Guennoun in favor of going through the project's preliminary guidelines. I learned more about the procedures followed by the Health Center's staff during those discussions. ATTENDANT was created based on the demands and specifications provided by the customer, the medical team and Dr. Guennoun. The planning stage of the development process takes up the first several weeks. I had the opportunity to collect all the necessary data during this phase on the requirements specifications and on the features needed in accordance with their procedures for handling students' records. A preliminary ERD that adheres to the defined business standards was also created using student paperwork and medical forms that I was equipped with at the start of the project. Dr. Guennoun outlined that Health Center is in need of a system that makes it easier to maintain students' data, and to handle medical forms, appointments with internal or external doctors, visit-consultations, and medication histories. The program must be scalable, secure, useful, and dependable.

2.1 Functional Requirements:

i.   *Student*

The nurse/ doctor should have the ability to modify the records of a students by :

- Add student

- See the list of students

- Search student

ii.   *Prescription*

The nurse/ doctor should have the ability to modify the records of a students by :

- Add Prescription

- View Prescription

- Search Prescription

- Delete Prescription

- edit Prescription

iii.   Student records

The nurse/ doctor manages the records of each student by:

- View records

- Edit record

- Delete record

- Explore CSV file of all the records

- Import CSV file file one and more records

iv.   Account

The admin should have the ability to register

- Create an account

- Reset the password by Admins

- login to account

- Delete an account by Admins

- Edit profile information

2.2 Non-Functional Requirements

i.   *Usability Requirement*

The ATTENDANT interface is simple and user friendly. It is easy to use and to adapt in the health sector. The functionalities are clear and quick to be accessed. The pages and the menu are in the same style and easy to navigate from one to another.

ii.   *Performance Requirement*

The software performs quickly and provides the output in a few seconds. The system should be responsive and provide quick responses to user actions. It should handle multiple concurrent users efficiently without significant delays or performance degradation.

iii.   *Development Requirement*

The system is built by using Laravel and its framework Livewire as a platform and mostly PHP as a backend programing language. Livewire helped me to add more simplicity to the project by using JavaScript code to perform the actions on the multiple form of ATTENDANT.

iv.   *Reliability Requirement*

The application should perform as required and expected. The system should be highly reliable, with minimal downtime and errors. It should be able to recover gracefully from failures and ensure data integrity at all times.

v.   *Portability Requirement*

The piece of code should operate with Windows operating systems. The system should be platform-independent, capable of running on different operating systems and hardware configurations without requiring significant modifications.

vi.   *Ethical Requirement*

The Health Center Student Record Management System should follow ethical rules and regulations of AUI HC like securing student's personal pieces of information and respecting their sensitive and private data.

vii.   *Delivery Requirement*

The Health Center Student Record Management System is to be delivered by the Start of January. During the testing phase of the application, ran the following tinker, Factory Command in order to create dummy records in the table of student and test the accuracy and efficiency of the functionalities of ATTENDANT.

```
php artisan tinker

Post::factory()->count(20)->create()
```

Figure 2.2.1: Command for Dummy Records

2.3 Requirements Specification:

The demands of Dr. Guennoun, who is considered as manager and at the same time the customer of the health center, are taken into consideration during the designing phase of the system ATTENDANT. Even during requirements and specifications phase, all pertinent data on the desired features were gathered. The customer precisely stated these aspects in great detail. Customers placed an emphasis on the desktop application's simplicity and use. They requested the system to manage all areas of the center's operations, including the management of patients, documents, prescriptions, and some medical histories. The papers I used in part to develop the criteria are included in the report's appendices. The paperwork Dr. Guennoun handed out that are utilized in the HC are among those documentations.

2.3.1 Feasibility Study

The AUI HC can gain from this initiative because it is incredibly beneficial. A session to the HC can be somewhere around five minutes to an hour, according to nurses at the HC. Somedays, the waiting room gets crowded because of the time the nurse is taking to fill the prescription of each student. The number of students served on time depends on what time of the day the visit was and on the season. Moving from using a paper-based forms to fill prescriptions and gather student's records, to such an enormous folder database standing in the heart of the room including all patients' records and histories is to improve both nurses' performance and students' experience. Consequently, replacing the present overhead with an electronic system will increase the effectiveness of the center. Furthermore, the waiting area may be enlarged to accommodate more patients with the deletion of the folder database, which uses up a significant

amount of space. The AUI HC will now be provided with a new tool that will help them improve their future decisions in order to match the data gathered.

Because I'm able to use resources and facilities based on university, database like PhpMyAdmin for example, to carry out the research, it will be reasonably inexpensive. Additionally, I made use of proper free development software tools like Laravel, Bootstrap…etc. However, the database of the HC will be used as they will be equipped by ATTENDANT. Because student's health is at risk, it is critical that the system be dependable and always accessible. For this reason, I adopted two roles when creating the authentication of the application, an admin, and a simple user with limited access. A project of this scale also needs upwards of three months to be completed, especially considering I am balancing four other courses at the same time. This investigation led me to the conclusion that the project is indeed doable with very specific requirements. I am certain that I took this project from a concept into an actuality with the direction and support of professor Nisar Naeem Sheikh and the motivation to produce a product that will truly be useful for others and help others need and facilitate their lives.

2.3.2 Schedule

The requirements specification for the project's first phase was developed through discussions with the Dr. Guennoun in order to define what "digitalizing their job" meant. The doctor acknowledged that the added responsibility of managing student records manually has made their job more stressful. It makes their regular services more difficult and ineffective.

Therefore, the primary goal of this project was to list the difficulties encountered, which are essentially caused by keeping a lot of information on paper. The goal is always to create a web application that will handle such records and involve everyone at the health center. Therefore, the functional and non-functional requirements made up most of the requirements' specifications for this report. The project's second phase, which consists of the feasibility study, then began. In order to demonstrate how the requirements were properly addressed, a thorough report of the prospective Engineering Standards, programming languages, and servers used will be provided in this step. The design and analysis phase, which includes the creation of an entity relationship diagram, a use case diagram, and examples of sequence diagrams, was completed before moving on to the implementation phase.

The design and analysis phase, which includes the creation of an entity relationship diagram, a use case diagram, and examples of sequence diagrams, were fulfilled before moving on to the implementation phase. Finally, there will be the implementation phase, which will be supported with screenshots of the web application's primary features as well as certain crucial sections of the code.

Last but not least, during the deployment phase, this application will assist in managing student data by keeping track of their visits and appointments for the HC. For the safety of the data, those services demand rigorous security. The application will also enable staff to submit a CSV file to the database of student records and contain a list of references to all student records. It will also adhere to the capstone project's shared standards and guidelines as well as the IEEE SRS templates.

3.   Design and analysis

The design and analysis phase of the ATTENDANT project focused on establishing a robust database schema to support the efficient handling of health center operations at Al Akhawayn University. This phase was crucial for ensuring that the application could manage student health records, appointments, prescriptions, and other related data effectively. The Entity-Relationship Diagram (ERD) serves as the backbone of the database design, illustrating the entities involved in the system, their attributes, and the relationships between them.

Entities and Attributes:

1. Student

   - Attributes StudentID (PK), FirstName, LastName, DateOfBirth, Gender, ContactInfo, MedicalHistory

   - Description: Represents the students at Al Akhawayn University. The StudentID serves as the primary key.

2. Appointment

   - Attributes: AppointmentID (PK), StudentID (FK), DoctorID (FK), Date, Time, Reason, Status

   - Description: Manages the scheduling of appointments. AppointmentID is the primary key. StudentID and DoctorID are foreign keys linking to the Student and Doctor entities, respectively.

3. Doctor

   - Attributes: DoctorID (PK), Name, Specialty, ContactInfo

   - Description: Stores information about doctors. DoctorID is the primary key.

4. Prescription

   - Attributes: PrescriptionID (PK), StudentID (FK), DoctorID (FK), Date, Medication, Dosage, Duration

   - Description: Contains details about prescriptions issued to students. PrescriptionID is the primary key, with StudentID and DoctorID as foreign keys.

5. MedicalRecord

   - Attributes: RecordID (PK), StudentID (FK), VisitDate, DoctorID (FK), Diagnosis, Treatment

- Description: Records details of student visits. RecordID is the primary key, with StudentID and DoctorID as foreign keys.

6. User

- Attributes: UserID (PK), Username, Password, Role

- Description:Manages user accounts for accessing the system. UserID is the primary key.

Connectivity and Relationships:

- Student to Appointment: One-to-Many. A student can have multiple appointments but each appointment is for a specific student.

- Doctor to Appointment: One-to-Many. A doctor can have multiple appointments but each appointment is associated with one doctor.

- Student to Prescription: One-to-Many. A student can have multiple prescriptions, each issued by a doctor.

- Doctor to Prescription: One-to-Many. A doctor can issue multiple prescriptions to different students.

- Student to MedicalRecord: One-to-Many. A student can have multiple medical records, documenting different visits to the health center.

- Doctor to MedicalRecord: One-to-Many. A doctor can be associated with multiple medical records, indicating their involvement in the treatment of students.

ERD Diagram Comprehensive Design:

The ERD diagram will visually depict the entities listed above, their attributes, and the relationships between them. The diagram will clearly show the primary keys (PK) and foreign keys (FK) to indicate the relationships. For instance, the StudentID in the Appointment, Prescription, and MedicalRecord entities will be marked as a foreign key (FK) linking back to the Student entity's primary key. Similarly, DoctorID in the Appointment, Prescription, and MedicalRecord entities connects to the primary key of the Doctor entity.

The relationships will be represented with lines connecting the entities, with symbols at each end to illustrate the nature of the relationship (one-to-one, one-to-many, or many-to-many). This comprehensive ERD design will facilitate understanding how data flows within the ATTENDANT system and ensure that the database structure supports all required operations efficiently and effectively.
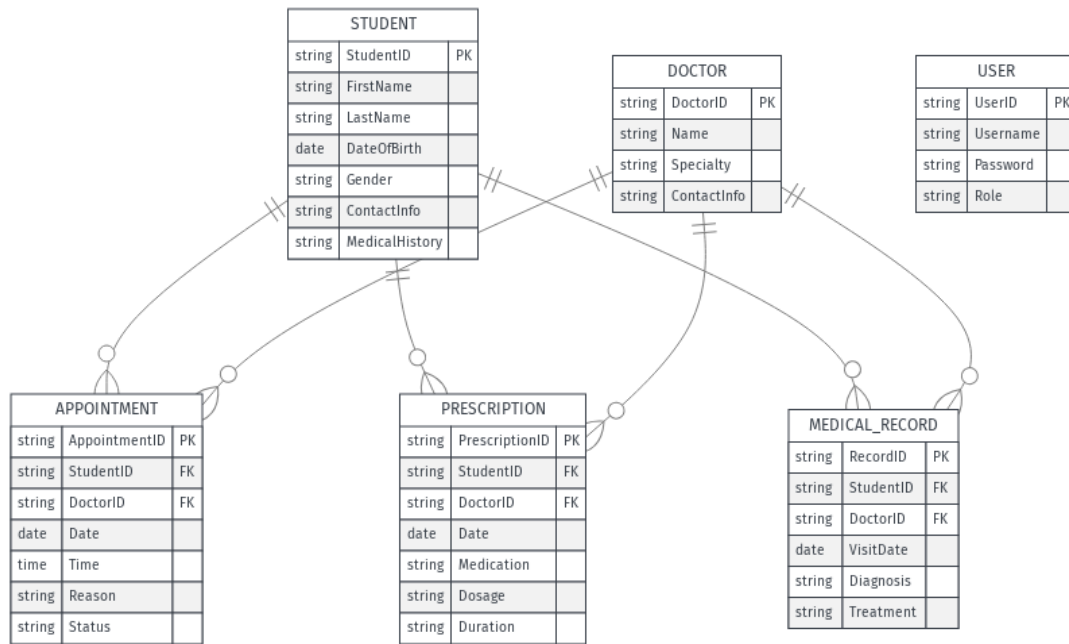
Figure 4.1: ERD

4.1 Use case diagram

A descriptive system diagram, the use case diagram shows the key attributes, elements, and functioning of the software. It displays the actors, use case, system, and many interrelationships. Conceptual Representation of the Use Case Diagram

- Actors:

  - Student: Positioned on the left side of the diagram.

  - Nurse/Doctor:Positioned on the right side, parallel to the Student actor.

  - Admin: Positioned on the right side, but slightly lower or above the Nurse/Doctor to indicate a different role

- System Boundary:

- Represented as a large rectangle labeled "ATTENDANT." All use cases are contained within this boundary.

- Use Cases:

- Manage Student Records:Placed near the Nurse/Doctor actor, as they are primarily interacting with this use case.

- Schedule Appointments: Centrally located between the Student and Nurse/Doctor actors, indicating interaction by both.

- Issue Prescriptions Close to the Nurse/Doctor actor, showing their responsibility for this function.

- View Medical Records:Accessible to both the Student and Nurse/Doctor, indicating shared use.

- User Account Management: Positioned near the Admin actor, highlighting their control over this area.

- Interactions:

- Lines connect each actor to their relevant use cases, illustrating the interactions. For example, a line from the Student to "Schedule Appointments" and "View Medical Records" use

cases; Nurse/Doctor to almost all use cases except "User Account Management"; and Admin specifically to "User Account Management.".

The diagram would visually depict the actors (Student, Nurse/Doctor, Admin) outside the system boundary, with lines connecting them to their respective use cases within the system. Each use case (Manage Student Records, Schedule Appointments, Issue Prescriptions, View Medical Records, User Account Management) would be represented as an oval within the system boundary, illustrating the system's functionality and how each actor interacts with it.

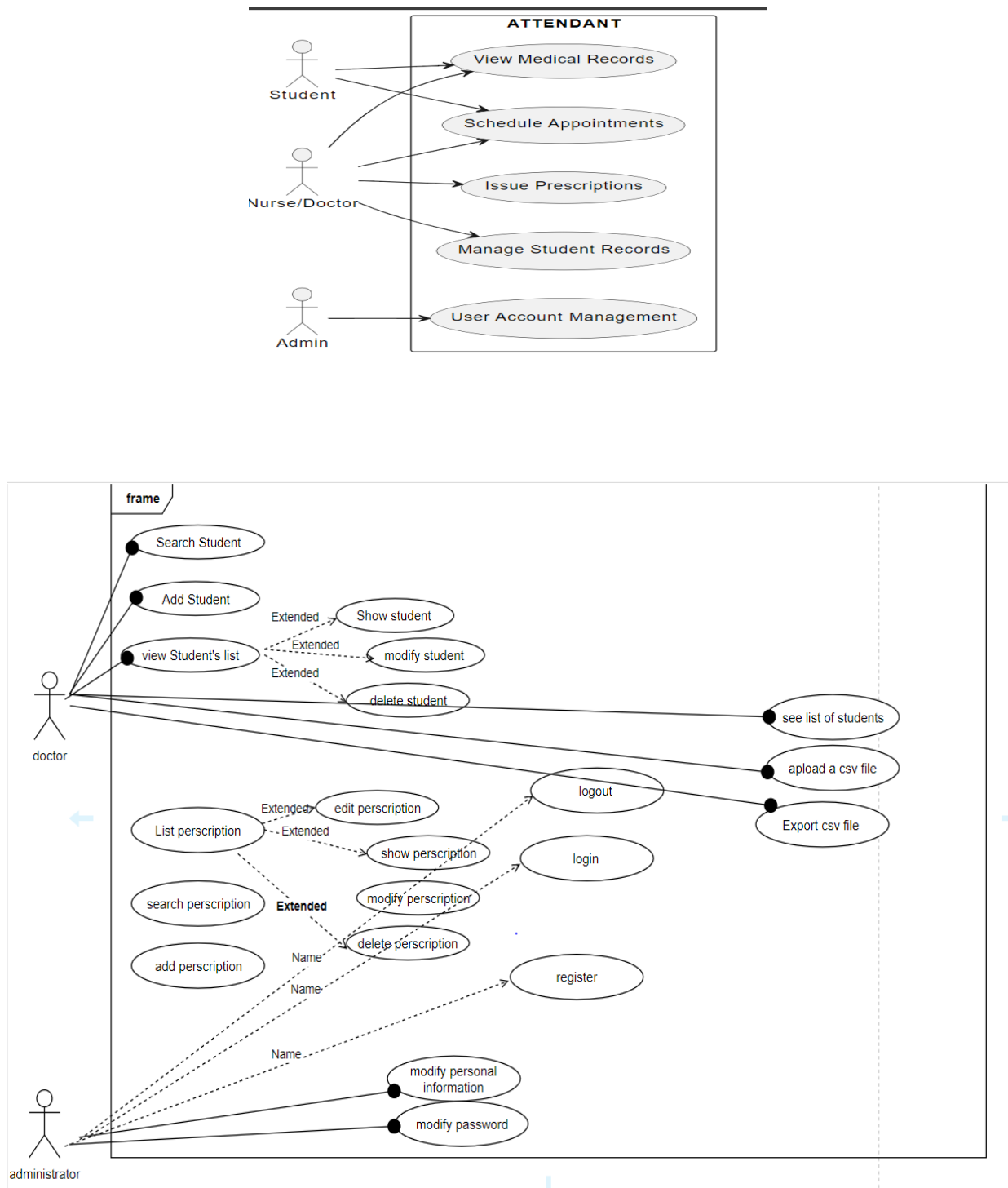Diagrams are shown in the figures below:

Figure 4.1.1: Sequence Diagram

The interactions between the various system items over a period of time are depicted in the sequence diagram. This diagram can be used as a flow model to record and validate the logic as well as for design and analytical reasons. [5] The Figure is a flowchart that shows how the authentication process unfolds.
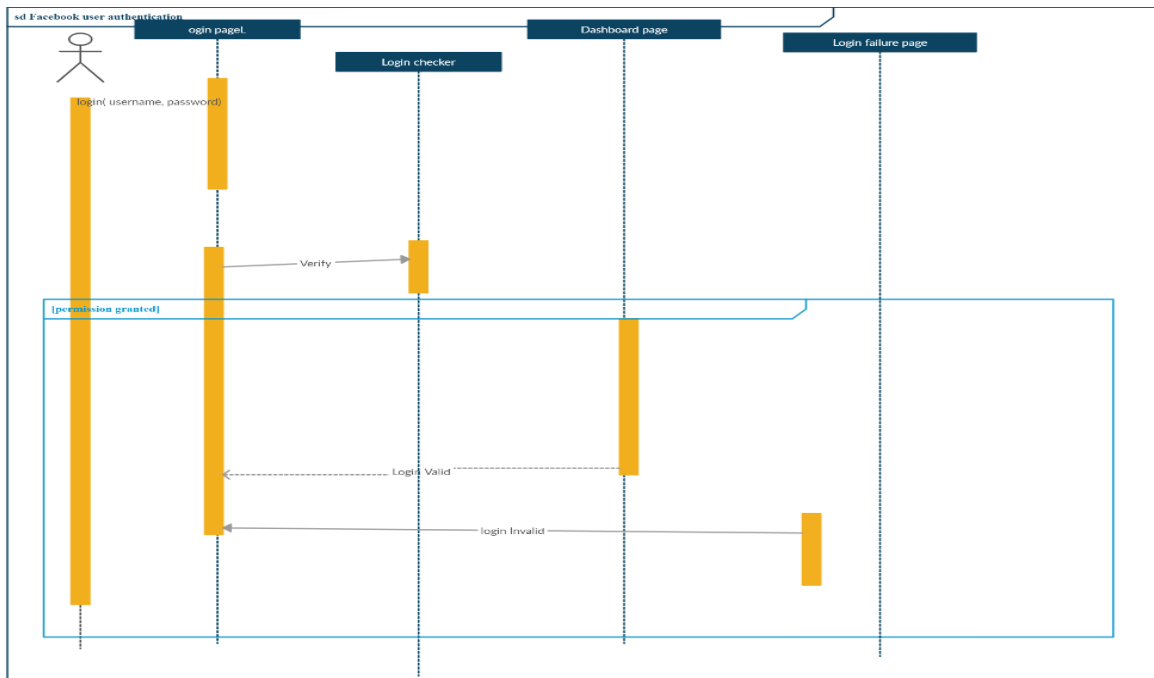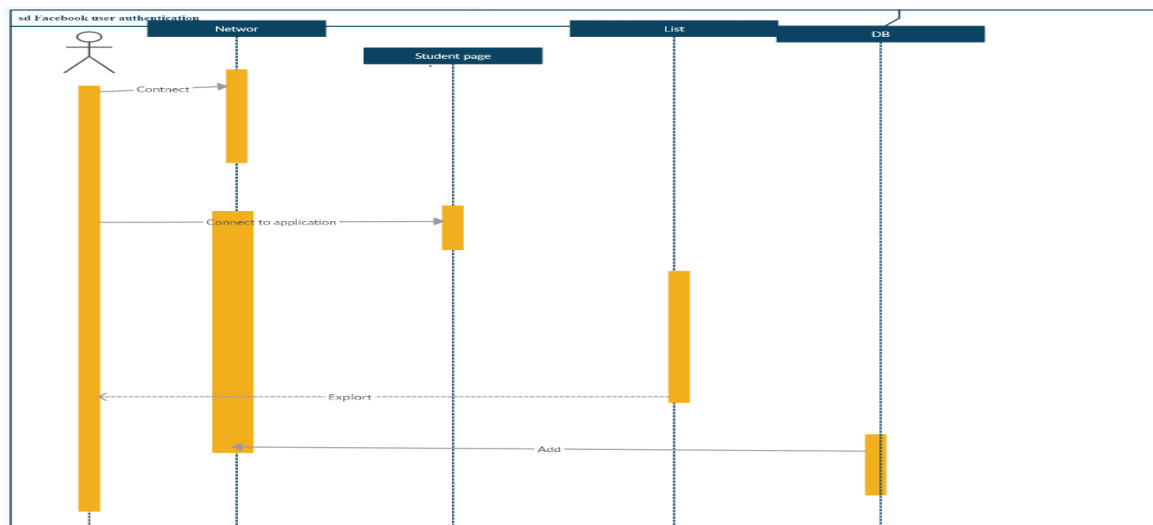

Figure 4.1.2: login

16

Figure 4.1.3: Students' records

The flow of adding a uploading and exploring student records in the students table as a csf file is modeled by the sequence diagram in figure.
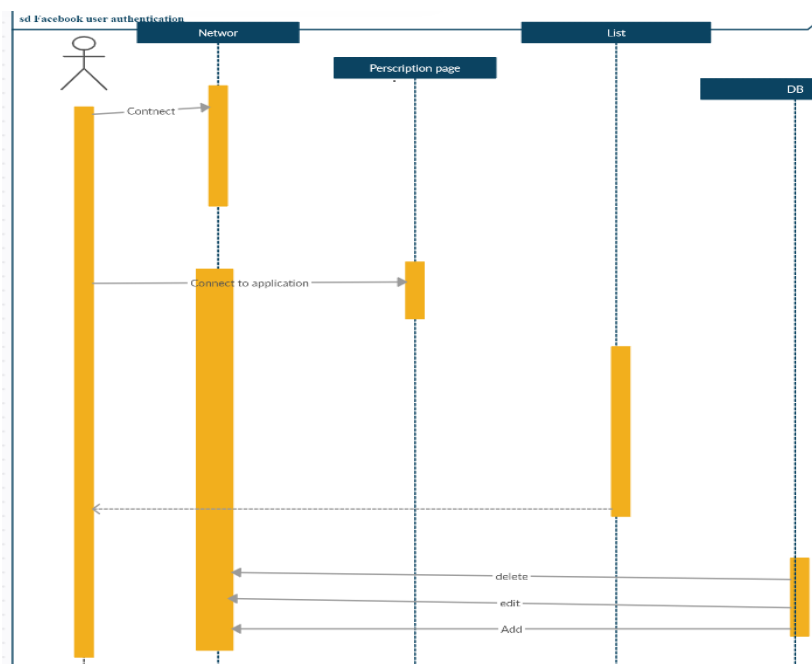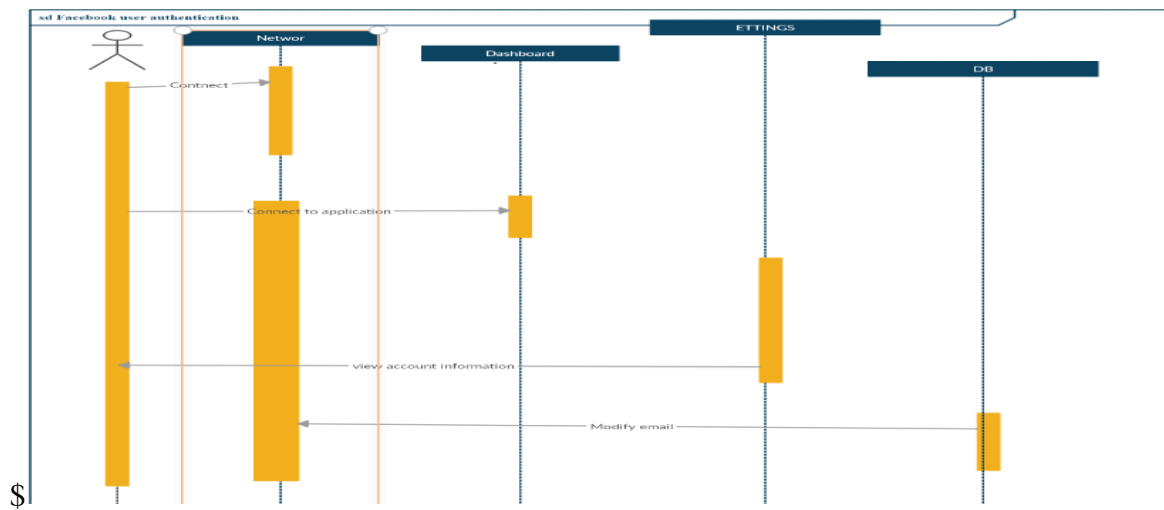

Figure 4 1.4: Prescription

17

Figure 4.1.5: Settings

4. STEEPLE Analysis

The STEEPLE analysis is a strategic planning method used by people and organizations to plan many components of this project with consideration to several external issues, including technological, political, social, ethical, economic, legal, and environmental. For this assignment, I used STEEPLE analysis to find the pertinent macroenvironmental elements. Given that the project is focused on a tiny organization—the staff health center—the steeple analysis is relatively condensed. The following list identifies the seven macroenvironmental elements that this project will consider:

5.1 Social Factor

At terms of the social impact, the ATTENDANT program will improve employee performance in the health center by automating their manual processes. Instead of wasting time and effort looking for medical papers on the shelf, the nurses may be given more practical medical-related activities to serve the needs of the students. Therefore, it will have an influence on the staff at the health Center in particular and the AUI Community as a whole.

5.2 Technical Factor:

The application won't have any technological effects on the cycle of innovation or the creation of any new tools. It is created with the use of real technology and instruments.

5.3 Environmental Factor:

The HC will use less paper on average thanks to the ATTENDANT, which will have a good effect on the environment.

5.4 Economic Factor:

No part of the organization's finances or employment will be impacted by the application. This factor is therefore unchanged.

5.5 Political Factor:

The application will adhere to the Health Center's rules. It doesn't engage in any political activity.

5.6 Legal Factor

Although managing this data carelessly might lead to a privacy infringement, the

application needs access to the students' private information in order to work effectively.

Otherwise, this component is not connected.

5.7 Ethical Factor

The user should have access to the data through this system. The interface must present

data that has been obtained from the database. So, ATTENDANT will also provide complete

access to the data to a small group of authorized individuals. As a result, this program is

completely moral.

5. Engineering Standards

I had to put the waterfall model's software development processes into practice in order to

complete my internship's requirements and goals. I had to initially learn new technologies and

tools for this project. At the beginning of the project, I quickly picked up the majority of them,

including Laravel, PHP, and Bootstrap. But after that, I was learning by putting time limitations

into practice. I primarily relied on documentation, YouTube videos, and tutorial campaigns for

this purpose. I used IEEE engineering standers.

6.1 Languages

JavaScript: For the front-end development we opted JavaScript. I was not familiar with the

language. So, it took me some time to learn about and get familiar with it. I as planning to

leannode.js but it was not feasible due to time limits. JavaScript is a computer language used on web pages.

```
    );
    Student::insert($values);
  // $this->reset();
//    $this->currentStep = 1;
$data = ['name'=>$this->family_name.' '.$this->given_name,'id_card_num'=>$this->id_card_num];
return redirect()->route('registration.success', $data);
    }
}
```

Figure 6.1. 1: JavaScript code in multiform file

HTML A computer language used on the internet is called HTML. Web pages are written in this language. The term "hypertext markup language," which is what the acronym stands for, means "hypertext markup language" in French. I created blade pages with it mostly for the front end of the Laravel framework's view session. Additionally, it has to be included into the Bootstrap codes.

```
<div class="card card-primary card-outline">
  <div class="card-body box-profile">
    <div class="text-center">
      <img class="profile-user-img img-fluid img-circle admin_picture" src="{{ Auth::user()->picture }}" alt="U
    </div>

    <h3 class="profile-username text-center admin_name">{{Auth::user()->name}}</h3>

    <p class="text-muted text-center">Admin</p>

    <input type="file" name="admin_image" id="admin_image" style="opacity: 0;height:1px;display:none">
    <a href="javascript:void(0)" class="btn btn-primary btn-block" id="change_picture_btn"><b>Change picture</b
```

Figure 6.1.2 HTML/CSS piece of code in Setting page

PHP HyperText Preprocessor, popularly known as PHP (self-referential acronym), is a free programming language that may be used to create any locally interpreted language as well as

21

dynamic Web pages via an HTTP server. I created Laravel framework controllers using PHP to govern how data is stored, edited, and deleted in the backend.

```html
<tr>

    <td>{{ $student->id }}</td>
    <td>{{ $student->given_name }}</td>
    <td>{{ $student->middle_name }}</td>
    <td>{{ $student-> family_name}}</td>
    <td>{{ $student-> gender}}</td>
    <td>{{ $student-> id_card_num}}</td>

    <td>{{$student->created_at }}</td>

@endforeach
```
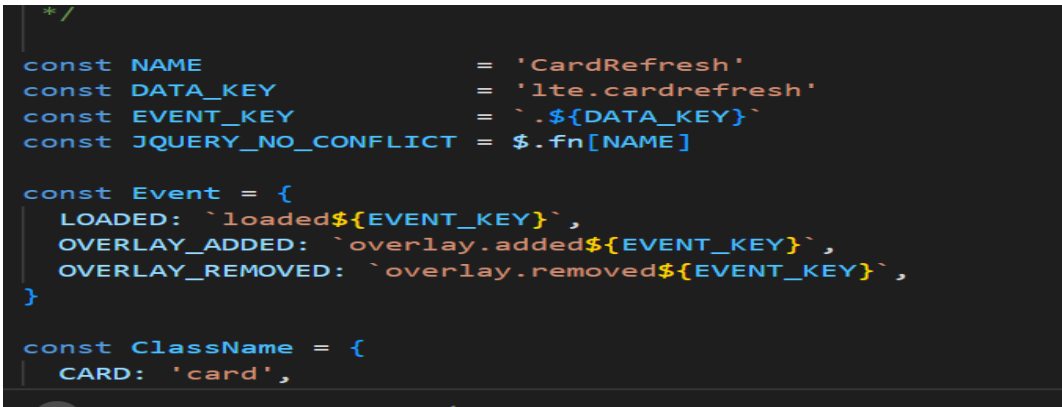
Figure 6.1.3: PHP piece of code

6.2 Framework

i.   *Bootstrap:*

I chose Bootstrap to increase usability and efficiency in front-end development. To enable the building of flexible pages, Twitter originally developed the open-source project known as Bootstrap (responsive web design). Developers may quickly construct applications that adapt to devices of different sizes and include common components like dialogs and validation thanks to its set of standard classes (which includes HTML and CSS codes). Thanks to Bootstrap, the application can be accessed on both desktop and mobile devices and is browser-based.

The usage of this Framework enabled the creation of pages that are appropriate for all display screen sizes and that dynamically adapt to the media format from which they are accessible (PC, tablet, Smartphone). I used the command Bootstrap to install it: php artisan ui bootstrap –auth.

```
*/

const NAME                  = 'CardRefresh'
const DATA_KEY              = 'lte.cardrefresh'
const EVENT_KEY             = `.${DATA_KEY}`
const JQUERY_NO_CONFLICT = $.fn[NAME]

const Event = {
  LOADED: `loaded${EVENT_KEY}`,
  OVERLAY_ADDED: `overlay.added${EVENT_KEY}`,
  OVERLAY_REMOVED: `overlay.removed${EVENT_KEY}`,
}

const ClassName = {
  CARD: 'card',
```

Figure 6..2.1: JavaScript code in bootstrap framework

ii.   *Laravel*

The model-view-controller (MVC) approach is adhered to by the open-source Laravel web framework, which was totally created using object-oriented programming. Taylor Otwell founded Laravel in June 2011. I utilized Laravel version 8 to complete the internship project's objectives. Because it is the most effective for facilitating collaboration between the front-end and the back end, I used the Laravel framework. I created this online application with it and had a wonderful full-stack experience, with the ability to employ a variety of web development tools including Ajax and Livewire. [1]. I used the package Fast Excel within Laravel that helped me work with the CSV file (explore and upload them) [8]

Include the PHP.

```
1    composer require livewire/livewire
```

Figure: 6.2 2. Livewire command


```php
public function export()
{
    return Excel::download(new StudentsExport, 'Students.xlsx');
}

/**
```

```
composer create-project --prefer-dist laravel/laravel:^9.0 laravel9-bootstrap5-vite
```

Figure: 6.2.3 Laravel instalment command

6.3 Library:

JQuery is a JavaScript library that is cross-platform, free, and designed to make it simpler to write client-side scripts in web pages' HTML code. JOHN RESIG released the initial version in January 2006. The entire set of fundamental functions for JQuery are included in a single JavaScript file that is 247 KB in size (92.2 KB in its minified version after removing comments, whitespace characters, and line breaks, and 32 KB with zip compression).

6.4 platform

i.   Microsoft Visual Studio

Microsoft Visual Studio is an IDE (Integrated Development Environment) that it developed

that is used to create desktop and web applications. Microsoft's SQL Server functionalities are

included in Visual Studio, making the creation of several functionalities (particularly when

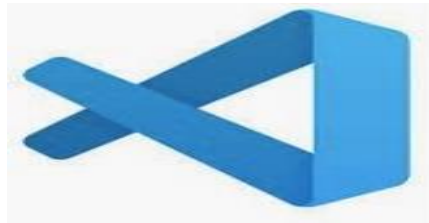querying the database) easier and more effective.

Figure: 6.3.1 Visual studio code

ii.   XAMP

Figure: 6.3.2. Xamp

7   Final Remarks

7.1 Challenges

This capstone project is one of the most projects I had to challenge myself and push the boundaries of I thought I was capable of. Indeed, it a was stressful, difficult, and challenging experience for me though out the semester. It demanded all the dedication, tenacity, and labor I could muster. Despite the fact that I chose to work with Laravel because it is reliable and easy to learn, I found difficulty meeting the requirements by operating some new web development techniques to add more usability and efficiency to the application. I decided to challenge myself and seize the chance to work with livewire and ajax.

The main difficulties I encountered while working on my capstone is related more to languages and technical terms. Furthermore, I had to use CRUD operations almost in every aspect of this project. It is true that I had some experience with Laravel and CRUD operation during my internship in Menara Holding, yet I seized the chance of working on a web application and implement Livewire. As much as livewire makes dynamic interfaces simple, it made it harder for me to implement CDUD operations since this is the first time using it, I had to learn about it and implement it to meet the requirements of my project which required more time and dedication.

Besides, I found it tough to change my mindset and logic of using pure Laravel and adapt to new Laravel frameworks. Some other difficulties I ran through were primarily time related. In truth, my original plan for this project was to be a hospital management system, but due to the Health Center's wide range of services and the time limits, I chose to concentrate primarily on

student record management. I chose this option because, as I've already stated, it is significantly more detailed and explicit and is feasible within the confines of a semester.

Along with taking four courses and a capstone, being an active student on campus made time management incredibly difficulties. I had to stay on campus during my spring break in order to continue working on this assignment as a result. I had to spend the entire semester learning how to code mostly in PHP and HTML because the application is more focused on the backend . Moreover, I had the opportunity to learn about the implementation of Excel and csv files with Laravel, the upload and export functionalities. However, the learning curve was shallow, and I was able to take up the languages and adapt to the framework fast because of the intense programming courses I took during my bachelor as a computer science student at Al Akhawayn University.

7.2 Lessons Learned

I believe I gained a lot of knowledge through this capstone assignment. It provided me with the opportunity to prove myself and bring the knowledge I obtained in my software engineering classes during my bachelor's degree studies at Al Akhwayn University into practice. Although challenging, the product development process was rewarding. When it comes to technical knowledge, I study and put into practice tools and technologies like Ajax livewire in Laravel and other languages like javaScript +visits, and their prescriptions. Within the dashboard, it is also possible to take care of the records of the doctors and nurses working in the Health Center and manage their information like their availability and their shifts. Indeed, it should go without saying that maintaining the application will be crucial, thus I concentrated on writing

relevant code to make this process easier. It won't be difficult for someone else to perform software maintenance if I am unable to do it myself.
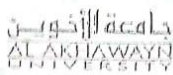
7.3 Conclusion

The management of student health data and the coordination of their interactions with the health center might be divided into two key sub-projects for this capstone project. But I can state with certainty that the first step took the longest amount of time because I had to utilize Livewire to build the form and connect it to the database. Since my initial objective for the application was to support the daily tasks of the Health Center employees, I was ready for this kind of result from the start. And it might necessitate adding additional features. That's why I decided to concentrate initially on Attendant's primary goal of record management. I believe that this project will be beneficial and have an effect on the offerings of the AUI Health Center. They will be able to quickly and simply manipulate data to serve kids. Filling out the initial form while registering for classes at AUI might also be helpful.

References

[1]. "Laravel - The PHP Framework For Web Artisans," *laravel.com*. https://laravel.com/docs/4.2/contributions.

[2]. "Laravel Tutorial - Tutorialspoint," *Tutorialspoint.com*, 2019. https://www.tutorialspoint.com/laravel/index.htm

[3] Laravel - The PHP Framework For Web Artisans," *laravel.com*. https://laravel.com/docs/9.x/errors.

[4] "Laravel Documentation," *GitHub*, Jan. 14, 2022. https://github.com/laravel/docs

[5]. Kelt Dockins, *Design patterns in PHP and Laravel*. Berkeley, Ca Apress, 2017.

[6] J. Lockhart, *Modern PHP*. "O'Reilly Media, Inc.," 2015.

[7]. "Laravel - The PHP Framework For Web Artisans," *laravel.com*.
https://laravel.com/docs/9.x/authentication

[8] "Class ExcelPackage," *epplussoftware.com*.
https://epplussoftware.com/docs/5.3/api/OfficeOpenXml.ExcelPackage.html

[9]. "Fast Excel Package for Laravel," *Laravel News*. https://laravel-news.com/fast-excel-package-for-laravel

[10] W3Schools, "W3Schools Online Web Tutorials," *W3schools.com*, 2019.
https://www.w3schools.com

[11] W3Schools, "HTML Reference," *W3schools.com*, 2019.
https://www.w3schools.com/tags/default.asp

Appendices

Appendix A: Patient Medical History Form

Allergies (if any) ☐ Yes ☐ No

a) Drugs (List drugs) _____

b) Other (food, bee stings, etc.) _____

Current and previous medical treatments:

Medical _____

Surgical _____

Attending Physician _____ Phone _____

Medical questionnaire

Do you have or have you ever had...?

Yes Oui	No Non		Yes Oui	No Non	
		Epilepsy or convulsion / épilepsie ou convulsions			Gastrointestinal Problems / Problèmes gastro-intestinales (ulcer, diarrhea, constipation)
		Diabetes / Diabète			Thyroid Problems / Problèmes de thyroïde
		Rheumatic Fever / Fièvre rhumatismale			High Blood Pressure / Hypertension
		Chronic Cough / Toux chronique			Drug Abuse / Abus de drogue
		Tuberculosis / Tuberculose			Headaches or migraines / Maux de tête ou migraines
		Respiratory Problems / Problèmes respiratoires (asthma, bronchitis)			Fainting / Évanouissements
		Deafness / Surdité			Bleeding Tendencies / Diathèses hémorragiques
		Visual Problems / Problèmes de yeux (Do you wear glasses?)			Tabacco use / Usage de tabac # of cigarettes per day _____ For how long? ____
		Alcohol use / Utilisation d'alcool # of drinks per week: _____ For how long? _____			Sexually Transmitted Disease / Maladies sexuellement transmissibles
		Gynecology problems / Problèmes gynécologiques			

Notes: _____

Evaluate your physical condition

☐ Below average ☐ Average ☐ Above average ☐ Excellent

Do you practice or have you played a sport? Yes ☐ No ☐

If Yes, which one: _____

Signature

I certify that the information given in this form is true and correct to the best of my knowledge. I understand that all this information will be kept confidential except for authorized medical personnel.

Signature _____ Date: _____

Appendix A: Patient Medical History Form