

# University Dental Clinic System

## System Requirements Specification

**COE 420 Software Engineering**

**American University of Sharjah**

**College of Engineering**

**Computer Science and Engineering**

Team Luigi
Joseph Press
Gregory Smith
Ayah Almheiri
Abdu Sallouh

## *Revision History*

Version	Date	Comment
1.0	4/10/2022	<ul style="list-style-type: none"><li>• 1st draft of SRS</li></ul>
1.1	10/10/2022	<ul style="list-style-type: none"><li>• Edited SRS based on initial feedback</li></ul>

## Contents

<b>1. Introduction</b>	<b>1</b>
1.1 Purpose	1
1.2 Scope of Project	1
1.2.1 Stakeholders	2
1.3 Overview	2
<b>2. Overall Description</b>	<b>3</b>
2.1 Product Perspective	3
2.2 Product Functions	3
2.3 User Characteristics	4
<b>3. External Interface Requirements</b>	<b>4</b>
3.1 User Interfaces	4
3.2 Hardware Interfaces	5
3.3 Software Interfaces	6
3.4 Communication Protocols Interfaces	6
<b>4. System Features</b>	<b>6</b>
4.1 Scheduling Appointments	6
4.1.1 Description	6
4.1.2 Action/Result	6
4.1.3 Functional Requirements	6
4.2 View and Edit Appointments	7
4.2.1 Description	7
4.2.2 Action/Result	7
4.2.3 Functional Requirements	7
4.3 Record Patient Visits and Create Medical Records	7
4.3.1 Description	8
4.3.2 Action/Result	8

4.3.3 Functional Requirements	8
4.4 Edit Availability	8
4.4.1 Description	8
4.4.2 Action/Result	8
4.4.3 Functional Requirements	9
4.5 View Work Schedule	9
4.5.1 Description	9
4.5.2 Action/Result	9
4.5.3 Functional Requirements	9
4.6 Price Dental Materials	9
4.6.1 Description	9
4.6.2 Action/Result	10
4.6.3 Functional Requirements	10
4.7 Calculate Visit Cost	10
4.7.1 Description	10
4.7.2 Action/Result	10
4.7.3 Functional Requirements	11
<b>5. Non Functional Requirements</b>	<b>11</b>
5.1 Product Non Functional Requirements	11
5.1.1 Usability	11
5.1.1.1 Device Compatibility	11
5.1.1.2 Learning Curve	11
5.1.1.3 Efficient Storage of Data	11
5.1.1.4 Self Explanatory Icons	11
5.1.2 Reliability	11
5.1.2.1 Resistant to Faults in Data	11
5.1.2.2 Reasonable Response Time	12

5.1.3 Security	12
5.1.2.1 Password Management	12
5.1.2.2 Secure Access	12
5.1.2.3 Secure Storage	12
5.1.3 Maintainability	12
5.1.3.1 System Downtime	12
5.1.3.1 System Maintenance	12
5.2 Process Non Functional Requirements	12
5.2.1 Design	12
5.2.1.1 Code Cleanliness:	12
5.2.1.2 Team Communications:	13
5.2.1.3 Backwards Compatibility:	13
5.2.2 Documentation	13
5.2.2.1 Standards for Deliverables:	13
5.3 External Non Functional Requirements	13
5.3.1 Legal Storage of Medical Data:	13
5.3.1 Interaction with University System:	13
<b>Appendix A: Glossary</b>	<b>13</b>
<b>Appendix B: References</b>	<b>14</b>

## 1. Introduction

Many transactions in the modern-day service industry involve some sort of appointment scheduling; this is especially true in a field such as health care, where a patient needs to be physically present to receive treatment. However, as a medical practice gets larger and the client base grows, scheduling appointments can become a nightmare for the overburdened receptionists. A myriad of factors need to be taken into consideration for a medical appointment system, including the distribution of clients with providers, the clients' schedules, and providers' existing appointments with other clients. This is likely why there has been a decisive shift towards automated appointment scheduling- a practice that is employed by many businesses.

**University Dental Clinic System (UDCS)** is meant to remove this burden from the employees. This intelligent system will handle the busy-work of scheduling appointments, storing/retrieving medical records, displaying schedules for the practitioners, and calculating costs of procedures, which will alleviate the employee workload. UDCS will, as outlined in this SRS document, provide a simple way for clients to register and schedule appointments, and it will handle the aforementioned backend work within the clinic.

### 1.1 Purpose

The purpose of the **System Requirements Specification (SRS)** document is to provide information regarding the implementation and design of the **University Dental Clinic System (UDCS)** in accordance with the requirements for capability. Namely, this software will allow patients to book an appointment online without needing to call the dental clinic, and it will distribute the work between the dental students, which will in turn streamline the efficiency of the clinic operations and the students' training.

### 1.2 Scope of Project

The **University Dental Clinic System (UDCS)** is designed to address the problem of using human labor to manage the scheduling and distribution of appointments, as well as the storage of patient records. This will be accomplished by a series of objectives, which include, but are not limited to, the following:

- Create a streamlined & easy-to-use interface to allow a patient to book an appointment
- Distribute the patient workload among the practicing dental students in an equitable manner
- Potentially allow for more patients to be assigned to weaker students who need more practice
- Provide a way to display work schedules to dentistry students

The UDCS shall provide the following capabilities:

- Log procedures that are administered during any given appointment, as well as the quantities of consumed materials
- Allow the receptionist to create a new medical record for a new patient when he/she arrives for the first visit
- Predetermine the price of medical care in advance, and provide a way for the patient to pay after the appointment (accounting for insurance deductions)
- Ensure that a patient is assigned to the same practicing dentist in future appointments

### 1.2.1 Stakeholders

The stakeholders in this system are the clinic's patients, the practicing dental students, the clinic support staff (including reception, admin, and accounting), and the university.

## 1.3 Overview

The overall objective of the **University Dental Clinic System (UDCS)** is to prevent the unnecessary manual workload on the employees, which will then allow them to enhance their overall work. In turn, these innovations carry with them the potential to increase productivity, annual revenue, employee morale, and customer satisfaction. This system will streamline employee responsibilities at the dental clinic so that staff members may fully concentrate on their own job without having to be concerned about appointments, storing patient records, etc.

Intuitive and reliable, UDCS will provide the dental staff members with the freedom they need to work more efficiently. Finally, it will solve the problem of inequitable distribution of work for the dental students by ensuring that there is an equal distribution of new patients being.

## 2. Overall Description

UDCS will provide an interface that has the goal of making the administrative work easier, which will yield a more polished experience for both the clinic employees and the clients. More specifically, the program will reduce time wasted from making appointments manually, handle the problem of uneven distribution of work among the practicing students, and store the records of the patients.

The portal has a number of features, including managing login accounts for dental students and patients, securely storing patient medical information, recording patient appointments & include data about the procedures that were administered, estimating the cost of each visit, and much more. Additionally, patients may make and modify their own appointments using a polished and easy-to-use portal, and dental students can log into the system to view their schedules.

### 2.1 Product Perspective

UDCS is designed to replace the current system that is in place for scheduling medical appointments. The traditional and outdated method whereby a phone call occurs between the client and the receptionist is no longer necessary; this process will be replaced with a product that, through satisfying the requirements outlined in this SRS document, defines a far more streamlined process wherein the customer can interact with an online interface to view available times and cross reference that with their own availability.

This product is to be designed from the perspective of expanding upon the already existing university system. More specifically, the UDCS will be reliant upon the university login interface, as it will be simply implementing that login window as an extension of the project to provide extra convenience for university students. This will allow the software to appear as another system within the university organization.

### 2.2 Product Functions

The objective of the portal includes, but not limited to:

- Patient medical records.
- Patient recorded visits.
- Login/Register account.



- Managing appointments.
- Prepare dental materials prices.
- Managing work schedules.
- Visit costs

## 2.3 User Characteristics

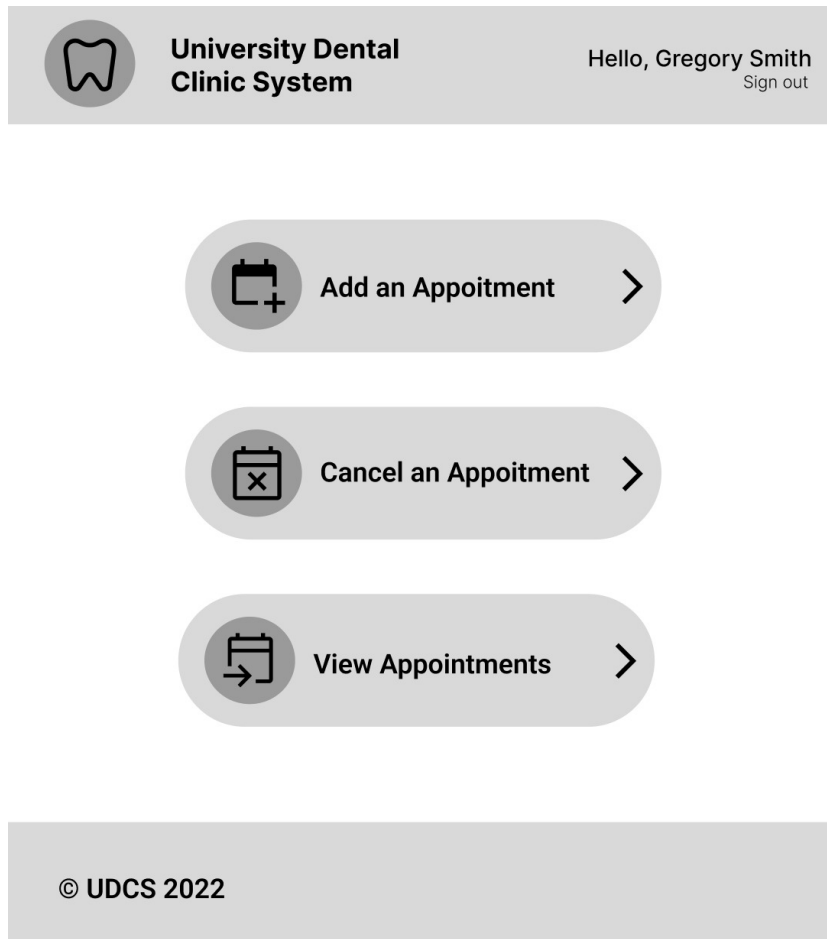
The usage of the UDCS is beginner-friendly and consistent, such that it can be relied upon to work as expected without hassle. Due to the system's straightforward functionality, the user may be able to master the minutiae in a matter of minutes.

The system will be split into two environments- one for the employees, and one for the patients. Each environment will allow the staff and patients to properly fulfill their respective roles in the transaction as outlined in the requirements (scheduling appointments, retrieving records, etc).

## 3. External Interface Requirements

### 3.1 User Interfaces

All users of the UDCS system will be provided with a web-based interface that will allow them to interact with the system within the bounds of their permissions. They will be prompted to log in, with the option of using the system that they are accustomed to using at the University or logging in manually. Once they have logged in, the following screen will appear differently, depending on whether the user is a patient, student dentist, administrative worker, receptionist, or accountant. Each category of user will have a clear way to perform their use cases. For example, a user who is a patient will have the option to either make a new appointment, edit an existing appointment, or view their existing appointment, in accordance with the use case diagram.



**Figure 3.1:** A sample patient interface offering options to add, cancel, and view appointments.

### 3.2 Hardware Interfaces

Being a web-based interface, the UDCS system is going to need to exist on a dynamically allocated cloud server so that a user can access the website through a search engine. The storage on the server must be dynamically allocated because the amount of data is expected to expand over time as more appointments are logged. Additionally, the server is going to be responsible for storage of information such as accounts, medical records, and appointments. Finally, the cloud server is going to be protected by industry standard security to ensure that user confidentiality is maintained.

### 3.3 Software Interfaces

This system will interface with the existing university sign in login system. Alternatively, the user will have the ability to login manually, but the option of logging in with university credentials will be presented for convenience.

### **3.4 Communication Protocols Interfaces**

The UDCS system will be hosted on a dynamically allocated server, which can be accessed through the HTTPS web protocol and wi-fi (the internet). This protocol will protect information as it travels from the server to the user's webpage, and it will be displayed on the .

## **4. System Features**

### **4.1 Scheduling Appointments**

One of the core features of the UDCS will be allowing a patient to book an appointment online.

#### **4.1.1 Description**

The UDCS will incorporate the feature of allowing the user to schedule an appointment for a medical visit at the hospital. Upon selecting to utilize the scheduler feature, the user can select a clinic in which to book an appointment. The user will be presented with an easy-to-understand calendar, where they will be able to select from available appointment times at that clinic from the screen. If the user has a history of seeing a certain provider, that student doctor's availability will be reflected in the available appointments.

#### **4.1.2 Action/Result**

The action of selecting an available appointment and confirming the selection will trigger the system to mark that time slot as booked.

#### **4.1.3 Functional Requirements**

- The user will have the option to cancel their appointment creation at any moment.
- The system will be up to date with the current status of appointment availability.
- The user will have the option of opting for a confirmation email containing their appointment details.
- The user will be able to input a reason for the appointment, or select from a list of common appointments (i.e. annual cleaning, fillings).

### **4.2 View and Edit Appointments**

In addition to being able to schedule appointments, the patient will also have the ability to view their upcoming appointments and cancel them as needed.

#### 4.2.1 Description

By selecting the option to view or edit appointments, the patient will be presented with a display of appointments that they have already created. Upon selecting an appointment from the list, they would be able to have the option to view further details or cancel the appointment. If they select to cancel the appointment, by clicking on a “save” button the change will be registered in the system.

#### 4.2.2 Action/Result

The action of selecting the view or edit appointments features will display the list of appointments. If, upon selecting an appointment from the list to view the details, the user decides to cancel the appointment and they save their changes, then the data will be updated within the system accordingly.

#### 4.2.3 Functional Requirements

- If the user selects to cancel an appointment, they should have the ability to cancel the operation by not selecting to save the changes.
- The appointments should be displayed to the user in the form of a simplified list, and clicking on the item should display a more detailed view of the appointment with the option to cancel.

### 4.3 Record Patient Visits and Create Medical Records

The receptionist will have access to the ability to record patient visits and, if applicable, create a medical record (which happens automatically if it is the patient’s first visit). This will be done through an easy-to-use interface that will allow them to enter the appointment details for a specific patient.

#### 4.3.1 Description

While filling out the details of an appointment record, the receptionist will query for the medical record of the patient who is being treated. If the patient is new and does not have a medical record, then a new record is created and the details of the visit are recorded there. Once all of the required information is filled, the receptionist will have the option to save it, which will commit the visit/medical record to the database.

### 4.3.2 Action/Result

Upon selecting to save the details, the cloud-based database will be promptly updated to reflect the addition to a patient's medical record.

### 4.3.3 Functional Requirements

- The receptionist will have the option to quit the creation of a record at any point in the process.
- While filling out the reason for the appointment, the receptionist will be presented with a list of common appointments (i.e. annual cleaning, fillings) to choose from, or they can write a custom appointment reason as needed.
- Once all of the fields are filled in, pressing "save" officially creates the patient record (and medical record if it is the patient's first visit).

## 4.4 Edit Availability

In order for a patient to be able to book an appointment with their dentist, they must be able to view the doctor's availability.

### 4.4.1 Description

The user-student dentist in this case- will be able to login to the system to update their availability. Upon logging in with their credentials and selecting the "Edit Availability" option, the dentist will be presented with a calendar interface, where they will be able to fill in their open appointment times in accordance with their schedule.

### 4.4.2 Action/Result

Once the dentist populates their availability, the changes will be immediately saved into the database. At this point, if a patient attempts to book an appointment with that dentist the change will be realized.

### 4.4.3 Functional Requirements

- Once an appointment is created by selecting an available time on their calendar, it will be assumed that this time slot is recurring weekly. If the student dentist would like to alter that appointment for a specific week, they will have the option to do so.

- Once an edit has been made there will be a button that allows the dentist to “save changes”. If they exit without selecting this button, then the changes to the schedule will be discarded.

## **4.5 View Work Schedule**

This module will allow the dentist student to see a list of their upcoming appointments, displayed on a calendar format.

### **4.5.1 Description**

It is in the best interest of the convenience of the student dentists to be able to quickly and easily see their updated work schedule. Selecting this function will present the student dentist with a calendar view that contains an up-to-date schedule of their appointments, and clicking on an appointment will display the details of the appointment.

### **4.5.2 Action/Result**

When the dental student selects that they would like to view their work schedule, a schedule in accordance with the SRS document will be displayed.

### **4.5.3 Functional Requirements**

- Each time the page is refreshed the calendar will be displayed with the most up-to-date values of the appointments, including indicating cancellations.
- Clicking on an appointment will display the details of the appointment that were entered by the customer when they created it.

## **4.6 Price Dental Materials**

This feature will allow an administrator to add or update prices for certain dental products.

#### 4.6.1 Description

The ability to add or adjust prices for dental materials is restricted to administrator access. An administrator with the proper credentials will have access to this function; selecting it will display a list of the current products and their prices. If the administrator decides to alter a price and select “save”, the new value will be updated in the database.

#### 4.6.2 Action/Result

Upon either 1) altering the price of a product or 2) adding a new product and setting a price, the user will be prompted to save the changes. If the user decides to proceed with saving the changes, the database will be updated with the new values, and these values will be used in the calculation for costs of procedures.

#### 4.6.3 Functional Requirements

- The current list of prices will be displayed in a list format, allowing the administrator to scroll through.
- The user will have the option to cancel the operation at any moment.
- Upon selecting to save the changes, the database will be updated with the new price values.

### 4.7 Calculate Visit Cost

During the dental visit, services and procedures will be provided to the patient and materials will be consumed. This system feature will allow clinics to keep track of the costs of these materials and bill patients for these expenses.

#### 4.7.1 Description

The prices of materials will be charged to the patient after the accounting department calculates the prices of these materials consumed. Only properly authenticated members of the accounting department will be able to access this function. The accounting department will not be able to change the prices of the materials set by the administration.

#### 4.7.2 Action/Result

The accounting department will add up the cost of the materials used in the visit. Then, the amount covered by insurance will be deducted from the total. If there is a remaining balance, the patient will pay that amount.

### 4.7.3 Functional Requirements

- Once the materials used during the visit are entered, the cost will automatically be calculated for presentation to the customer at the accounting department.
- The system will format a summary of the charges in a presentable format for the patient that include all of the materials, service(s), and insurance deductible.

## 5. Non Functional Requirements

### 5.1 Product Non Functional Requirements

#### 5.1.1 Usability

##### *5.1.1.1 Device Compatibility*

The various system interfaces should be compatible with most modern browsers, including (but not limited to) being able to accept connections from Chrome, Edge, and Firefox.

##### *5.1.1.2 Learning Curve*

The UDCS system will be easy to learn for any and all users. The employees shall be able to master the system in a few hours, while the average patient shall be able to understand their portion of the system in a matter of minutes.

##### *5.1.1.3 Efficient Storage of Data*

The data stored in the database shall be stored with a well-chosen data structure, causing response times to be low even as the size of the system grows over time.

##### *5.1.1.4 Self Explanatory Icons*

Any icons that are presented in the system will be reasonably understandable in accordance with their assigned purpose.

#### 5.1.2 Reliability

##### *5.1.2.1 Resistant to Faults in Data*

The data that is stored in the database shall contain backups that are maintained frequently, such that the system can quickly recover in the event of a data error.



### ***5.1.2.2 Reasonable Response Time***

UDCS shall have a reasonably fast response time, loading the webpage and any subsequent queries within a second.

## **5.1.3 Security**

### ***5.1.2.1 Password Management***

User passwords shall be stored as hashes in the server, not in plaintext, in order to preserve the security of the user credentials.

### ***5.1.2.2 Secure Access***

The website shall conform to basic modern networking standards to ensure a secure connection with the end user so that data cannot be intercepted while en route to the server.

### ***5.1.2.3 Secure Storage***

The server that contains the client data shall be stored in a manner that renders it extraordinarily difficult to compromise (i.e. encrypted) in order to preserve patient confidentiality.

## **5.1.3 Maintainability**

### ***5.1.3.1 System Downtime***

In the case of a failure, the system should be able to be restored within a single business day.

### ***5.1.3.1 System Maintenance***

Regular system maintenance shall be promulgated ahead of time so that the user can be aware of when to not access the service.

## **5.2 Process Non Functional Requirements**

### **5.2.1 Design**

#### ***5.2.1.1 Code Cleanliness:***

The codebase of the system shall be properly documented in order to allow future engineers to be able to easily debug

#### **5.2.1.2 Team Communications:**

The development team shall meet on a weekly basis to run diagnostic checks and confirm that everything is working as intended.

#### **5.2.1.3 Backwards Compatibility:**

Future software updates shall be designed to be backwards compatible with previous versions in order to allow for downgrades in case of an issue.

### **5.2.2 Documentation**

#### **5.2.2.1 Standards for Deliverables:**

Future project deliverables shall be presented in a format similar to this SRS document, closely following the IEEE Std. 830(1984).

## **5.3 External Non Functional Requirements**

### **5.3.1 Legal Storage of Medical Data:**

The database shall conform to standards of data protection in accordance with government directives.

### **5.3.1 Interaction with University System:**

The user interface should reliably integrate with the University Login system.

## **Appendix A: Glossary**

Term	Definition
<b>Calendar View</b>	<b>Calendar View</b> is a commonly used way of displaying data as it relates to dates/events. Common examples include Google Calendar and Apple Calendar.
<b>Data Structure</b>	A <b>data structure</b> is a way of storing data. Different data structures are better suited for storing different amounts or types of data.
<b>Feature</b>	A software <b>feature</b> is a capability that satisfies a requirement, specifically in pertinence to those outlined in this SRS document.
<b>Interface</b>	An <b>interface</b> (or <b>graphical user interface</b> ) is a way for a user to interact with a program.

<b>Protocols</b>	<b>Protocols</b> enable communication across the internet.
<b>Querying</b>	A <b>query</b> is a search based on input criteria, particularly through a database of information.
<b>Stakeholder</b>	<b>Stakeholders</b> are people who have an interest in a product's usability, longevity, and success in addressing problems.
<b>Web-based Interface</b>	The ability for a user to connect to code running on a remote server through the user of a web browser (such as Chrome, Firefox, and Edge).

## Appendix B: References

Al-Zinati, Mohammad. "SRS-Sample 1". *American University of Sharjah, Software Engineering Project*. 2013.

"Data Protection Laws." *Data Protection Laws - the Official Portal of the UAE Government*,  
<https://u.ae/en/about-the-uae/digital-uae/data/data-protection-laws#:~:text=It%20prohibits%20the%20processing%20of,the%20legal%20procedures%20and%20rights>.

Gibson, Ben, et al. "Data Protection and Cybersecurity Laws in UAE: CMS Expert Guide." *Data Protection and Cybersecurity Laws in UAE | CMS Expert Guide*,  
<https://cms.law/en/int/expert-guides/cms-expert-guide-to-data-protection-and-cyber-security-laws/uae>.