

AD HEALTH

Healthcare Appointment Management Database



A project by:
Damian Rozycki
Anh Phan

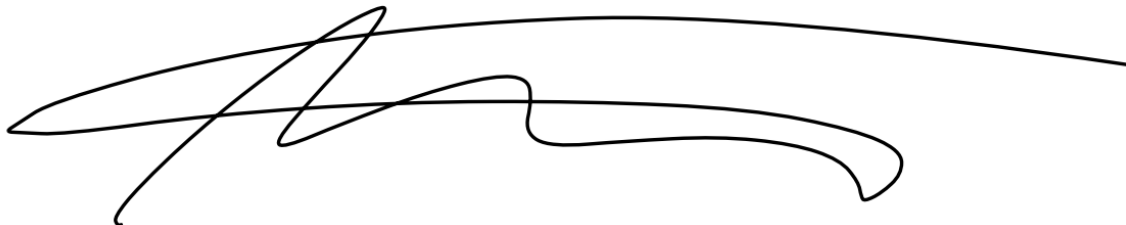
Towson University
Spring 2024

COSC 457 – Database Management Systems

Table of Contents

1.	<i>INTRODUCTION</i>	2
1.1	Project Overview	2
1.2	Purpose of the Database.....	2
1.3	Data Integrity and Security	2
1.4	Supporting System Functionality	2
1.5	Scalability and Flexibility	3
1.6	Conclusion	3
2.	<i>TARGET BUSINESS</i>	3
3.	<i>BUSINESS PROCESS</i>	4
3.1	Sequences of business activities	4
3.2	Process Modeling	6
3.3	Data Model Components.....	6
4.	<i>USER REQUIREMENTS</i>	7
4.1	Entity Relationship Data Model	7
4.2	Functional Requirements	8
4.3	Database Queries	8
5.	<i>ENTITY RELATIONSHIP DIAGRAM (ERD)</i>	9
6.	<i>GANTT CHART FOR THE PROJECT</i>	10

Signature of interviewee



4/29/24

1. INTRODUCTION

1.1 Project Overview

The AD Healthcare Appointment Management System is a groundbreaking project aimed at modernizing appointment scheduling and management in healthcare facilities. At its core, this project seeks to create a robust database that supports the system's functionality, ensuring seamless interaction between users and the platform.

1.2 Purpose of the Database

The primary purpose of this database design is to facilitate the storage, retrieval, and management of critical data related to appointment scheduling, patient information, and healthcare provider availability. By creating a structured and efficient database, the AD system aims to streamline operations for healthcare facilities and improve the overall patient experience.

1.3 Data Integrity and Security

One of the fundamental objectives of this database design is to ensure data integrity and security. Patient information, appointment schedules, and provider availability are sensitive data that require strict measures to safeguard confidentiality. The database will implement robust security protocols to protect against unauthorized access and data breaches.

1.4 Supporting System Functionality

The database design will encompass a range of critical data points essential for the smooth operation of the AD Healthcare Appointment Management System. These include, but are not limited to:

1. User Accounts (Doctors, Patients, Managers, and Admins)
2. Patient Information
3. Insurance Information
4. Medical History
5. Branch information
6. Department information
7. Doctor Information
8. Appointment Information

Each of these features relies on a well-structured database schema to store and retrieve data efficiently, providing users with a seamless and user-friendly experience.

1.5 Scalability and Flexibility

As the AD system expands and evolves, the database design is built with scalability and flexibility in mind. It is designed to accommodate growth in data volume, user base, and additional functionalities. This ensures that the system can adapt to changing needs and demands within the healthcare industry.

1.6 Conclusion

In conclusion, this database design aims to provide a secure, efficient, and scalable solution for healthcare facilities to manage appointments and improve patient care. By focusing on data integrity, security, and supporting system functionalities, this database design sets the stage for a modern and innovative healthcare management platform.

2. TARGET BUSINESS



- **Name of Business:** Maryland Vascular Specialists
- **Branches:**

Maryland Vascular Specialists - Locations

Outpatient Centers & Vascular Labs

Cockeysville Center

Glen Burnie Center

Rosedale Center

Hagerstown Center

Offices & Vascular Labs

Pikesville Office

Westminster Office

Belvedere Square Office

Laurel Office

Columbia Office

Abingdon Office

Essex Office

Havre De Grace Office

Bel Air Office

Offices & Vascular Labs In PA

York Office

Hanover Office

Chambersburg Office

- **Size of Business:** Medium
- **Business activities:**
 - Services: Wound Care, Vascular Treatments, Vascular Surgery, Diagnostic Lab
 - Target area: Maryland/Pennsylvania/DC

3. BUSINESS PROCESS

3.1 Sequences of business activities

The following scenarios outline the sequences of activities related to the new database application, from user registration to post-appointment procedures.

1. User Registration and Account Creation:

- Scenario: A new patient visits the Maryland Vascular Specialists website.
 - Action: The patient clicks on the "Register" button.
 - Action: The patient fills out the registration form with their personal information.
- Result: The system creates a new user account for the patient.

2. Location Search and Service Selection:

- Scenario: The registered patient needs to schedule an appointment.
 - Action: The patient logs into their account.
 - Action: The patient enters their location preference and desired service.
- Result: The system displays available clinic locations and services.

3. Appointment Booking with Healthcare Provider:

- Scenario: The patient selects a specific doctor for an appointment.
 - Action: The patient chooses a date and time slot for the appointment.
- Result: The system confirms the appointment and sends a notification to the patient.

4. Patient Treatment by Healthcare Provider:

- Scenario: The patient arrives and the provider begins the treatment process.
 - Action: The doctor accesses the patient's record and updates it with treatment details.
- Result: The system logs the treatment details and updates the patient's record.

5. Post-Appointment Procedures:

- Scenario: After the appointment, the provider updates the patient's progress.
 - Action: The provider logs into the system to input post-appointment notes.
- Result: The system records the post-appointment information for future reference.

3.2 Process Modeling



3.3 Data Model Components

RELATIONS with PRIMARY KEY (PK) and FOREIGN KEY (FK):

1. MVS_BRANCH:

- Branch_No (PK)

2. DEPARTMENT:

- Department_No (PK)
- Branch_No (FK)
- Manager_ID (FK)

3. USER:

- User_ID (PK)
- User_Type_ID (FK)
- Manager_ID (FK)

4. USER_TYPE:

- User_Type_ID (PK)

- Description (FK)

5. MANAGER:

- User_ID (PK)
- Department_Supervised_ID (FK)

6. DOCTOR:

- User_ID (PK)
- Department_No (FK)

7. DOCTOR_AVAILABILITY:

- Availability_ID (PK)
- Doctor_User_ID (FK)

8. NOTIFICATIONS:

- Notification_ID (PK)
- Patient_User_ID (FK)
- Doctor_User_ID (FK)

9. PATIENT:

- User_ID (PK)

10. ALLERGY:

- Allergy_ID (PK)
- Patient_User_ID (FK)

11. MEDICATION:

- Medication_ID (PK)
- Patient_User_ID (FK)

12. MEDICAL_RECORDS:

- Record_ID (PK)
- Patient_User_ID (FK)

13. INSURANCE:

- Insurance_Card_No (PK)
- Patient_User_ID (FK)
- Primary_Doctor_User_ID (FK)

14. APPOINTMENT:

- Appointment_ID (PK)
- Patient_User_ID (FK)
- Doctor_User_ID (FK)

4. USER REQUIREMENTS

4.1 Entity Relationship Data Model

Maryland Vascular Specialists (MVS) has many **BRANCHes**, which are uniquely identified by a Branch_no, Branch_name, and Location. Each BRANCH is divided into multiple **DEPARTMENTS** (with unique Dept_no and Dept_name).

MVS has many PATIENTs, DOCTORs, and MANAGERs; each has their own **USER** account with a unique User_ID, followed by a Username, Password, Email, Name (First, Middle, Last), DOB, Sex, SSN, Phone_#, Email, Address (Street, City, State, Zip_Code) and Date_Registered. Each USER account is assigned a specific **USER_TYPE** with a User_Type_ID and a Description; for example, '1' is for 'PATIENT', '2' is for 'DOCTOR', and '3' is for 'MANAGER'.

Each DEPARTMENT at the MVS is supervised by a **MANAGER**. MANAGERs will have a Start_Date, End_Date, Office_No, and Salary. They can also make changes to doctors' work schedule and manage **DOCTOR_AVAILABILITY** (Start_time, End_time, Date, Is_Available).

PATIENTs will have a Height, Weight, **ALLERGIES**, and **MEDICATIONS**. It is required for each PATIENT to have an **INSURANCE** (with Card_no, Company_Name, Company_Phone_#, Expiry_date, and Primary_Doctor_User_ID). A PATIENT may also have a **MEDICAL_RECORD** in the system, including Record_ID, Description, and Date.

A **PATIENT** can make an **APPOINTMENT** with any **DOCTOR**, and each **APPOINTMENT** is identified by an **Appt_ID**, **Location**, **Start_time**, **End_time**, and **Date**.

Each **DOCTOR** will work for a specific **DEPARTMENT** and have information like **License_#**, **Specialization**, **Years_of_Experience**, and **Salary**. **DOCTORs** can receive **NOTIFICATIONS** (**Date**, **Time**, **Message**, **Status**) about **PATIENTs** and **APPOINTMENTs**.

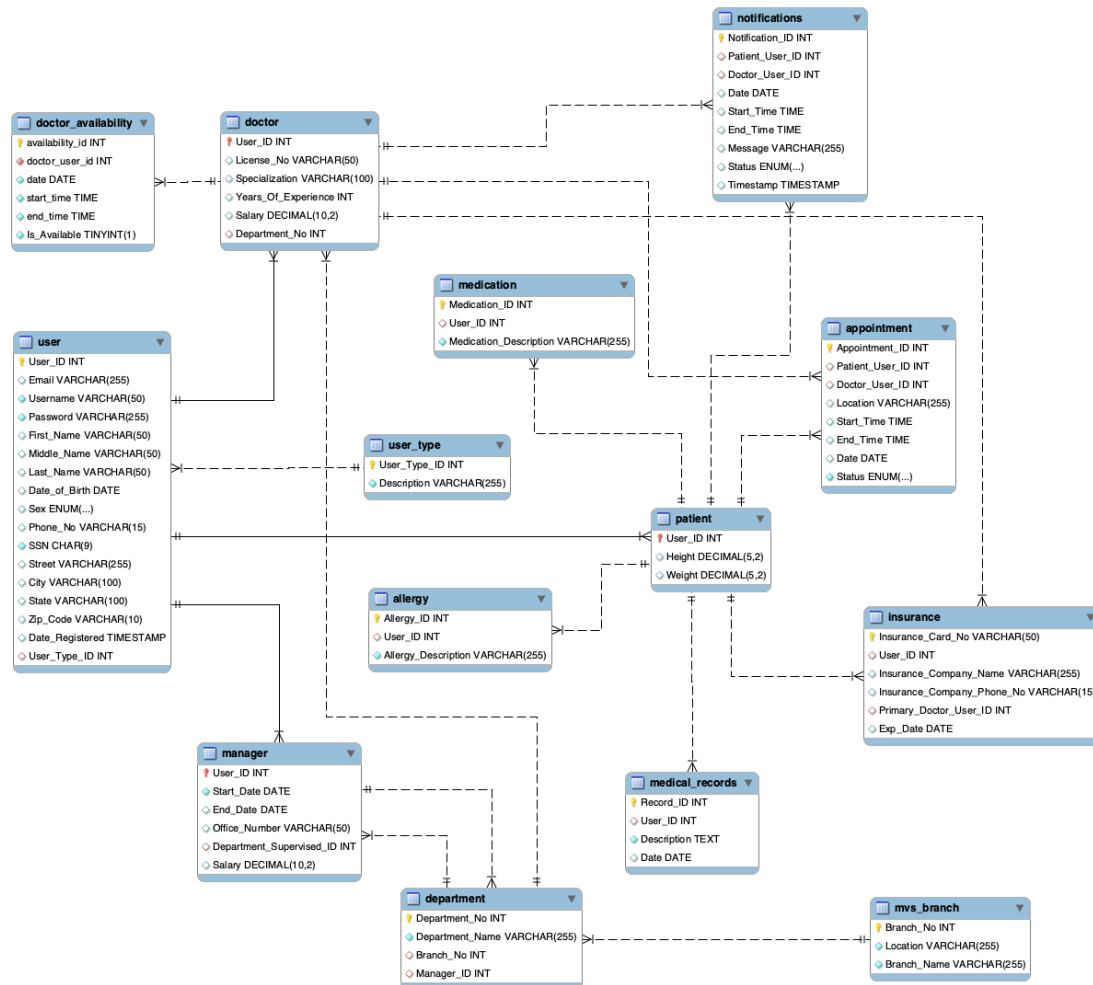
4.2 Functional Requirements

1. Allow Patients to view / book / update / cancel appointments.
2. Allow Patients to view / add / update / delete information in their medical records.
3. Allow Doctors to retrieve patient information and medical history.
4. Allow Doctors to update patient treatments and medical records.
5. Allow Doctors to view / create / update / delete appointments.
6. Allow Managers to view / add / update / delete patient and doctor information.
7. Allow Managers to make changes doctors' schedule and availability.

4.3 Database Queries

1. Retrieve the first name, last name of a patient.
2. Retrieve the first name, last name of a doctor.
3. Retrieve the salary of a specific doctor.
4. Retrieve the total salary of an entire department.
5. Retrieve the start date, ssn of all managers working for any department from any branch.
6. Retrieve all doctors along with their specialties and department information.
7. Retrieve the name of medications that a specific patient has.
8. Retrieve all appointments for a specific patient with provider details.
9. Retrieve the total number of staff and patients of all branches.
10. Retrieve all doctors first name, last name, salary with more or less than a specific number of patients treated in the last year.
11. Retrieve all patients who have the same insurance provider.
12. Retrieve the first and last names of all providers who have worked more than a certain number of years.
13. Retrieve the essn, first name, last name of all managers who supervise wound care departments in all branches of MVS.
14. Retrieve the total number of patients who have had an appointment with a doctor who works in a specific department of a specific branch.
15. Retrieve the first name, last name of all managers who work in a specific branch.

5. ENTITY RELATIONSHIP DIAGRAM (ERD)



Assumptions:

- A patient may have multiple allergies or medications.
- A manager supervises at most one department.
- Each department can only have one manager.
- Each doctor works for at most one department.

6. GANTT CHART FOR THE PROJECT

Tasks	Start date	End date	Duration
Phase 1			
Research Organization	3/12	3/13	6 hours
Finalize Organization	3/13	3/13	1 hours
Prepare Questions for interviewee	3/15	3/15	4 hours
Conduct Interview	3/16	3/16	1 hours
Analyze Requirements	3/17	3/17	1 hours
Design ERD	3/20	3/20	3 hours
Construct ERD	3/22	3/22	2 hours
Review Possible queries	3/22	3/22	1 hours
Revise ERD	3/23	3/24	2 hours
Ensure ERD meets requirements	3/25	3/25	1 hours
Finalize ERD	3/25	3/25	30 minutes
Create Schema	3/25	3/25	1 hours
Finalize Schema	3/29	3/29	3 hours
Write up Gantt chart	3/29	3/29	1 hours
Submit Project report 1	4/3	4/4	30 minutes
Phase 2			
Review project 2 requirements	4/29	4/29	1 hours
Design Database in SQL	4/30	5/1	2 days
Create Database using SQL	5/2	5/2	1 day
Review and revise Database	5/3	5/3	1 day
Design UI in Java SWING	5/4	5/6	3 days
Conduct Testing	5/7	5/7	5 hours
Revise Database and UI	5/7	5/7	3 hours
Finalize Database Design and UI Design	5/8	5/8	5 hours
Create PPT slides	5/8	5/8	4 hours
Practice Presenting	5/8	5/8	2 hours
Present in class	5/9	5/9	20 minutes
Adding more functions to the application	5/9	5/11	3 days
Finalize Database and UI Design	5/12	5/12	1 day
Update report 1	5/12	5/12	1 day
Create a flyer	5/13	5/13	3 hours
Make a tutorial manual	5/13	5/14	2 days
Finalize all documents	5/14	5/14	1 day
Submit Final Project 2	5/14	5/14	10 minutes