

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN



# BÁO CÁO ĐỒ ÁN THỰC HÀNH

Môn: Cơ sở dữ liệu nâng cao

**Giảng viên hướng dẫn:**

Cô. Hồ Thị Hoàng Vy

Cô. Tiết Gia Hồng

TP Hồ Chí Minh, ngày 26 tháng 12 năm 2023

## MỤC LỤC

I.	GIỚI THIỆU .....	4
1.	Thông tin nhóm:.....	4
2.	Công cụ hỗ trợ:.....	4
II.	THÔNG TIN ĐỒ ÁN.....	4
1.	Tên sản phẩm: .....	4
2.	Về đồ án: .....	4
3.	Mục tiêu chính của báo cáo:.....	4
III.	PHÂN TÍCH ĐÓNG GÓP CÁ NHÂN.....	6
1.	Công việc được giao:.....	6
2.	Đánh giá từng thành viên:.....	6
IV.	MÔ TẢ QUY TRÌNH VÀ CHỨC NĂNG.....	6
1.	Phân tích quy trình phòng khám nha khoa:.....	6
2.	Mô tả quy trình phòng khám nha khoa:.....	7
3.	Mô tả chức năng:.....	7
V.	THIẾT KẾ CƠ SỞ DỮ LIỆU MỨC QUAN NIỆM VÀ LOGIC.....	8
1.	Mô tả Bảng và chi tiết thuộc tính trong mô hình:.....	8
2.	Xác định các quan hệ thể hiện trong mô hình:.....	10
3.	Mô hình dữ liệu mức quan niệm: .....	13
4.	Lược đồ quan hệ: .....	14
5.	Phân tích phụ thuộc hàm - dạng chuẩn: .....	14
6.	Mô hình dữ liệu mức logic: .....	16
VI.	THIẾT KẾ CƠ SỞ DỮ LIỆU MỨC VẬT LÝ .....	16
1.	Biên dịch thiết kế cơ sở dữ liệu ở mức vật lý:.....	16
1.1.	Thiết kế bảng cơ sở: .....	16
1.2.	Tạo trigger quản lý dữ liệu khi nhập vào bảng:.....	28
1.3.	Thiết kế business rules:.....	30
2.	Nối các phần tiền trình thực hiện câu lệnh với các bảng: .....	31
3.	Ánh xạ các tiền trình vào bảng: .....	33
4.	Phi chuẩn hóa: .....	35
VII.	CẢI THIỆN HIỆU QUẢ TRUY VẤN .....	35
1.	Phân tích hiệu năng: .....	35
2.	Thiết lập chỉ mục (index): .....	36
3.	Đề xuất tối ưu hóa truy vấn: .....	37

3.1.	Kỹ thuật phân trang: .....	37
3.2.	Cải thiện câu Select:.....	39
4.	Các hàm được sử dụng để truy vấn dữ liệu trong ứng dụng: .....	39
VIII.	DEMO ỨNG DỤNG .....	49
1.	Chức năng chính trong ứng dụng: .....	49
2.	Validation trong ứng dụng:.....	68
IX.	TÀI LIỆU THAM KHẢO .....	70

## I. GIỚI THIỆU

### 1. Thông tin nhóm:

NHÓM 11			
	Mã số sinh viên	Họ và tên	Email
1	21127235	Nguyễn Xuân Quỳnh Chi	<a href="mailto:nxqchi21@clc.fitus.edu.vn">nxqchi21@clc.fitus.edu.vn</a>
2	21127495	Lê Ngô Song Cát	<a href="mailto:lnscat21@clc.fitus.edu.vn">lnscat21@clc.fitus.edu.vn</a>
3	21127659	Bùi Ngọc Kiều Nhi	<a href="mailto:bnknhhi21@clc.fitus.edu.vn">bnknhhi21@clc.fitus.edu.vn</a>

### 2. Công cụ hỗ trợ:

Tên đồ án	HỆ THỐNG QUẢN LÝ PHÒNG KHÁM NHA KHOA
Công cụ hỗ trợ	Visual Studio Code
	XAMPP – PHP
	MSSQL Server
	Jira – Github

- **Github:** <https://github.com/BNKieuNhi/AdvDB-DentalClinic.git>

## II. THÔNG TIN ĐỒ ÁN

### 1. Tên sản phẩm:

Hệ thống quản lý phòng khám nha khoa.

### 2. Về đồ án:

- Ngày nay, sự tiện lợi và hiệu quả của việc sử dụng hệ thống quản lý trong nhiều nghiệp vụ doanh nghiệp không chỉ giúp cho người dùng nâng cao trải nghiệm mà còn giúp các doanh nghiệp dễ dàng hơn trong việc thống kê, quản lý, tối ưu hóa quy trình làm việc. "Hệ thống Quản lý Phòng khám Nha khoa" là một ứng dụng website phục vụ cho việc quản lý các hoạt động của phòng khám nha khoa. Qua đó, việc quản lý hồ sơ bệnh nhân, nhân viên, và nha sĩ trở nên dễ dàng hơn.
- Ứng dụng website này cho phép đặt lịch hẹn cho bệnh nhân dựa trên thông tin lịch rảnh của nhiều nha sĩ thuộc phòng khám, và các thông tin liên quan đến đơn thuốc sau khi kết thúc lần khám. Ngoài ra, các nha sĩ sẽ thuận lợi hơn trong việc kê đơn thuốc và dịch vụ bệnh nhân cần sử dụng. Việc sử dụng website giúp người dùng dễ dàng truy cập xem thông tin từ nhiều thiết bị khác nhau và đảm bảo tính khả dụng cao. Song song đó, hệ thống giúp cải thiện tính hiệu quả và tiết kiệm thời gian cho phòng khám trong việc quản lý, ghi nhận thông tin.

### 3. Mục tiêu chính của báo cáo:

- Mô tả chi tiết quy trình và chức năng:
  - Tập trung vào việc mô tả chi tiết về quy trình và chức năng của hệ thống phòng khám nha khoa. Điều này bao gồm các bước và quy trình mà người dùng sẽ thực hiện khi sử dụng hệ thống, cũng như chức năng cung cấp bởi hệ thống cho người dùng.
- Thiết kế dữ liệu mức quan niệm và logic:

- Định nghĩa các thực thể (entities), thuộc tính (attributes) và mối quan hệ (relationships) giữa chúng.
- Mục tiêu là tạo ra một mô hình dữ liệu chính xác và phù hợp với yêu cầu và logic của hệ thống.
- Thiết kế cơ sở dữ liệu mức vật lý:
  - Xác định cấu trúc và định dạng cụ thể của bảng bao gồm kiểu dữ liệu, chỉ mục, ràng buộc, khóa chính và khóa ngoại.
  - Mục tiêu là tạo ra một cơ sở dữ liệu hiệu quả, có hiệu suất tốt và đáp ứng được yêu cầu của hệ thống.
- Cải thiện hiệu quả truy vấn:
  - Đề xuất cách tối ưu hóa cấu trúc truy vấn, sử dụng chỉ mục, tối ưu hóa câu lệnh truy vấn và các kỹ thuật khác để cải thiện hiệu suất truy vấn.
  - Mục tiêu là đảm bảo rằng hệ thống có thể xử lý các truy vấn nhanh chóng và hiệu quả.

### III. PHÂN TÍCH ĐÓNG GÓP CÁ NHÂN

#### 1. Công việc được giao:

##### QLPKNK Sprint 1 30 Oct – 26 Nov (10 issues)

Mô tả chi tiết quy trình, dữ liệu, ràng buộc liên quan. Danh sách chức năng cần xây dựng kèm tần suất giao dịch tương ứng (tự quan sát, phân tích và đề xuất). Thiết kế dữ liệu mức quan niệm và logic.

		CONCEPTUAL DESIGN	DONE	① 01 NOV	4	L
✓ QLPKNK-4	Giới thiệu đồ án, mô tả chi tiết quy trình hoạt động	CONCEPTUAL DESIGN	DONE	① 01 NOV	4	L
✓ QLPKNK-5	Liệt kê danh sách chức năng cần xây dựng	CONCEPTUAL DESIGN	DONE	① 03 NOV	4	L
✓ QLPKNK-6	Mô tả thực thể chính cùng với các thuộc tính liên quan	CONCEPTUAL DESIGN	DONE	① 07 NOV	6	N
✓ QLPKNK-7	Xác định mối quan hệ giữa các thực thể	CONCEPTUAL DESIGN	DONE	① 07 NOV	6	N
✓ QLPKNK-8	Xây dựng mô hình dữ liệu mức quan niệm	CONCEPTUAL DESIGN	DONE	① 12 NOV	6	BN
✓ QLPKNK-9	Mô tả lược đồ quan hệ, phân tích phụ thuộc hàm, dạng chuẩn	LOGICAL DESIGN	DONE	① 15 NOV	5	N
✓ QLPKNK-10	Xây dựng mô hình dữ liệu mức logic	LOGICAL DESIGN	DONE	① 15 NOV	6	BN
✓ QLPKNK-11	Phác thảo giao diện website	WEBSITE DESIGN	DONE	① 15 NOV	8	L
✓ QLPKNK-12	Thiết kế giao diện	WEBSITE DESIGN	DONE	① 20 NOV	10	BN
✓ QLPKNK-13	Lập trình giao diện hệ thống	WEBSITE DESIGN	DONE	① 26 NOV	12	BN

+ Create issue

##### QLPKNK Sprint 2 27 Nov – 29 Dec (10 issues)

Xây dựng bảng thiết kế cơ sở dữ liệu mức vật lý theo đề xuất của nhóm, các đề xuất về cải thiện hiệu quả truy vấn dựa trên thiết kế đề xuất, source code của project mô phỏng các chức năng, source script tạo csdl, query, index...

		PHYSICAL DESIGN	DONE	① 01 DEC	8	BN
✓ QLPKNK-14	Biên dịch, thiết kế cơ sở dữ liệu ở mức vật lý	PHYSICAL DESIGN	DONE	① 01 DEC	8	BN
✓ QLPKNK-15	Ánh xạ các tiến trình vào bảng	PHYSICAL DESIGN	DONE	① 06 DEC	8	N
✓ QLPKNK-16	Phân tích hiệu năng	PHYSICAL DESIGN	DONE	① 06 DEC	6	L
✓ QLPKNK-17	Thiết lập chỉ mục	PHYSICAL DESIGN	DONE	① 10 DEC	8	N
✓ QLPKNK-18	Tối ưu hóa truy vấn	PHYSICAL DESIGN	DONE	① 12 DEC	8	L
✓ QLPKNK-19	Tạo dữ liệu ảo	PHYSICAL DESIGN	DONE	① 10 DEC	8	L
✓ QLPKNK-21	Lập trình chức năng Login-Logout, Update, Delete	WEBSITE DESIGN	DONE	① 16 DEC	14	L
✓ QLPKNK-22	Lập trình chức năng Add	WEBSITE DESIGN	DONE	① 17 DEC	10	N
✓ QLPKNK-23	Lập trình chức năng View, Search	WEBSITE DESIGN	DONE	① 17 DEC	10	BN
✓ QLPKNK-24	Demo các chức năng, validation của hệ thống	WEBSITE DESIGN	DONE	① 21 DEC	4	BN

+ Create issue

#### 2. Đánh giá từng thành viên:

	Mã số sinh viên	Họ và tên	Đánh giá mức độ hoàn thành
1	21127235	Nguyễn Xuân Quỳnh Chi	100%
2	21127495	Lê Ngô Song Cát	100%
3	21127659	Bùi Ngọc Kiều Nhi	100%

### IV. MÔ TẢ QUY TRÌNH VÀ CHỨC NĂNG

#### 1. Phân tích quy trình phòng khám nha khoa:

- Phòng khám nha khoa sẽ bao gồm các quy trình : bệnh nhân đặt lịch hẹn, nha sĩ điều trị và hẹn lịch tái khám, nha sĩ chỉ định lộ trình điều trị cho bệnh nhân, nha sĩ kê đơn thuốc cho bệnh nhân, bệnh nhân thanh toán hóa đơn điều trị và thuốc.

## 2. Mô tả quy trình phòng khám nha khoa:

- Quy trình bệnh nhân đặt lịch hẹn:  
Đặt lịch hẹn khám :
  - + Khách hàng liên hệ nhân viên để cung cấp thông tin cơ bản và ngày, giờ, phòng và nha sĩ sẽ khám dựa trên nha sĩ rảnh vào thời gian đó. Thêm vào đó, khách hàng có thể lựa chọn ghi chú thêm những thông tin đặc biệt về tình trạng dị ứng hay chống chỉ định thuốc của bản thân.
- Quy trình điều trị :  
Thêm kế hoạch điều trị:
  - + Sau khi thăm khám, nha sĩ sẽ tiến hành cập nhật thông tin về tình trạng sức khỏe răng miệng của bệnh nhân.
  - + Sau đó, nhân viên sẽ tiến hành chọn bác sĩ và danh mục điều trị cho bệnh nhân gồm chọn liệu trình, chọn răng và bề mặt răng cần chữa trị bằng liệu trình đó.
  - + Mỗi kế hoạch điều trị sẽ có trạng thái khác nhau bao gồm : kế hoạch, đã hoàn thành, đã hủy.
- Quy trình kê đơn thuốc cho bệnh nhân:
  - + Nha sĩ sẽ thực hiện kê những loại thuốc cần thiết cho bệnh nhân sau ngày điều trị đó.
- Quy trình thanh toán:
  - + Nhân viên tạo hóa đơn cho một kế hoạch điều trị gồm các thông tin về ngày lập hóa đơn, giá tiền của các liệu trình, giá tiền thuốc và tổng số tiền khách hàng phải trả.
  - + Khách hàng sẽ thực hiện thanh toán hóa đơn.
  - + Sau khi khách hàng đã thanh toán, nhân viên/ quản trị viên sẽ thêm vào trường hình thức thanh toán, tiền đã trả, và tiền thối lại cho khách hàng.

## 3. Mô tả chức năng:

- Quản lý tài khoản:
  - + Đăng nhập bằng tài khoản đã có : Người dùng có thể trải nghiệm các chức năng có trên hệ thống bằng việc nhập tên đăng nhập và mật khẩu tương ứng.
  - + Cập nhật thông tin tài khoản : Hệ thống cho phép người dùng cập nhật các thông tin cá nhân như họ tên, địa chỉ, số điện thoại, giới tính, ngày sinh và mật khẩu.
  - + Khóa tài khoản : Tài khoản đã bị khóa sẽ không thể đăng nhập vào website. Quản trị viên là người có quyền khóa tài khoản.
- Quản lý cuộc hẹn và hồ sơ bệnh nhân:
  - + Đặt cuộc hẹn : Khách hàng có thể thực hiện đặt cuộc hẹn thông qua nhân viên trực hệ thống bằng cách cung cấp các thông tin cá nhân và về ngày, giờ, phòng của cuộc hẹn và nha sĩ sẽ khám dựa trên nha sĩ rảnh vào thời gian đó.
  - + Ghi chú thông tin đặc biệt: Khách hàng có thể lựa chọn cung cấp cho nhân viên thêm những thông tin đặc biệt về tình trạng dị ứng hay chống chỉ định thuốc của bản thân.
  - + Nhân viên có quyền tạo hồ sơ bệnh nhân nếu bệnh nhân mới đến đặt cuộc hẹn.
  - + Hiển thị lịch hẹn cho nha sĩ : Nha sĩ sẽ được coi những cuộc hẹn của mình trong từng ngày.
- Quản lý số phòng khám : Trong một khung giờ, mỗi phòng khám chỉ có một cuộc hẹn diễn ra.

- Quản lý việc điều trị:
  - + Cập nhật tình trạng sức khỏe răng miệng cho bệnh nhân: Nha sĩ sau khi thăm khám sẽ ghi nhận lại kết quả của bệnh nhân vào hồ sơ bệnh nhân.
  - + Lên kế hoạch điều trị : Sau khi chẩn đoán, nha sĩ sẽ lên kế hoạch những liệu trình điều trị bệnh nhân cần làm qua việc chọn tên liệu trình dựa trên danh mục liệu trìn sẵn có và chọn răng cần để điều trị.
- Quản lý đơn thuốc:
  - + Kê đơn thuốc cho bệnh nhân: Nha sĩ có quyền thực hiện tạo đơn thuốc cho bệnh nhân thông qua hệ thống.
  - + Thuốc khách hàng bị chống chỉ định thì bác sĩ sẽ không thể kê cho bệnh nhân.
- Quản lý hóa đơn và thanh toán:
  - + Thanh toán: nhân viên sẽ tiến hành in hóa đơn từ hệ thống gồm ngày giao dịch, tổng tiền khách hàng cần trả dựa trên tổng chi phí dịch vụ đã điều trị và thuốc đã được kê cho bệnh nhân.
  - + Lựa chọn phương thức thanh toán: có hai hình thức thanh toán là tiền mặt và thanh toán online. Khi thanh toán hóa đơn, nhân viên lựa chọn phương thức thanh toán.
  - + Tính toán tiền thối lại cho khách hàng: đối với phương thức thanh toán bằng tiền mặt, hệ thống sẽ tính số tiền nhân viên cần thối lại cho khách hàng dựa trên số tiền nhận được và tổng tiền nhân viên cần trả.
- Quản lý lịch làm việc của nha sĩ: Quản trị viên được phép xem/thêm/cập nhật thông tin nha sĩ và nhân viên trong phòng khám cũng như thêm lịch làm việc của từng nha sĩ. Nha sĩ và nhân viên được phép xem lịch làm việc và danh sách làm việc của các nha sĩ.
- Quản lý nhân sự: Khi có nhân viên hoặc nha sĩ, quản trị viên là người sẽ được quyền thêm nhân sự mới và danh sách nha sĩ và nhân viên sẽ được cập nhật và hiển thị.
- Quản lý thuốc: Quản trị viên được phép xem/thêm/xóa/cập nhật thuốc có trong danh mục thuốc.

## V. THIẾT KẾ CƠ SỞ DỮ LIỆU MỨC QUAN NIỆM VÀ LOGIC

### 1. Mô tả Bảng và chi tiết thuộc tính trong mô hình:

#### 1.1. Bảng ACCOUNT

Thuộc tính	Mô tả
Username	Tên đăng nhập vào website
Password	Mật khẩu của tài khoản
isActive	Trạng thái của tài khoản. Nếu là “Yes” thì tài khoản đang được sử dụng, “No” thì tài khoản đã bị khóa/ngưng sử dụng.

#### 1.2. Bảng USER (bao gồm Admin, Staff, Dentist)

Thuộc tính	Mô tả
ID_User	Mã định danh người dùng
Fullname	Họ và tên của người dùng
Gender	Giới tính của người dùng
PhoneNumber	Số điện thoại liên lạc của người dùng

Address	Địa chỉ của người dùng
UserType	Phân biệt loại người dùng gồm : “Admin”, “Dentist”, “Staff”

**1.3. Bảng CUSTOMER**

Thuộc tính	Mô tả
ID_Customer	Mã định danh khách hàng
Fullscreen	Họ và tên của khách hàng
Gender	Giới tính của khách hàng
PhoneNumber	Số điện thoại liên lạc của khách hàng
Address	Địa chỉ của khách hàng
DOB	Ngày sinh của khách hàng

**1.4. Bảng APPOINTMENT**

Thuộc tính	Mô tả
ID_Appointment	Mã định danh cuộc hẹn
Date	Ngày diễn ra của cuộc hẹn
Time	Thời gian diễn ra của cuộc hẹn
Status	Trạng thái của cuộc hẹn gồm : “New”, “Reassess”.

**1.5. Bảng ROOM**

Thuộc tính	Mô tả
ID_Room	Mã định danh phòng thuộc phòng khám
Floor	Tầng của phòng khám

**1.6. Bảng TOOTH**

Thuộc tính	Mô tả
ID_Tooth	Mã định danh của răng
ID_Surface	Mã định danh của bề mặt răng
ToothName	Miêu tả tên và vị trí của răng : mặt trong răng cửa, mặt ngoài của răng khôn,..

**1.7. Bảng TOOTH\_PROBLEM**

Thuộc tính	Mô tả
CustomerID	Mã định danh khách hàng dùng để xác định vấn đề về răng thuộc về khách hàng nào.
Decription	Mô tả rõ hơn về vấn đề nha khoa.
NotedDate	Ngày và giờ ghi nhận vấn đề nha khoa.

**1.8. Bảng MEDICINE**

Thuộc tính	Mô tả
ID_Medicine	Mã định danh của thuốc
MedicineName	Tên của thuốc

Price	Giá tiền hiện tại của thuốc
-------	-----------------------------

**1.9. Bảng TREATMENT**

Thuộc tính	Mô tả
ID_Treatment	Mã định danh của liệu trình điều trị
TreatmentName	Tên của liệu trình điều trị
Description	Mô tả cụ thể về liệu trình điều trị đó.

**1.10. Bảng SELECT\_TREATMENT**

Thuộc tính	Mô tả
ID_Select	Mã định danh cho một lựa chọn điều trị cho khách hàng
DateSelect	Ngày và thời gian lựa chọn điều trị
ReturnDays	Số ngày cần để khách hàng quay lại tái khám.
SelectionStatus	Trạng thái của kế hoạch điều trị, bao gồm 3 trạng thái: “Planning”, “Completed”, “Canceled”.
Note	Những ghi chú cần thiết trong mỗi lựa chọn điều trị.

**1.11. Bảng INVOICE**

Thuộc tính	Mô tả
ID_Invoice	Mã định danh của hóa đơn (Số hóa đơn)
DentalPrice	Tổng số tiền dịch vụ mà khách hàng đã sử dụng.
MedicineFee	Tổng tiền thuốc khách hàng được nha sĩ kê.
Total	Tổng số tiền mà khách hàng phải thanh toán.
AmountPaid	Tiền khách hàng đã đưa.
Change	Tiền thối lại cho khách hàng.
InvoiceTime	Thời gian khách hàng thực hiện thanh toán hóa đơn

**1.12. Bảng PAYMENT\_METHOD**

Thuộc tính	Mô tả
ID_Payment	Mã định danh phương thức thanh toán của khách hàng gồm : “Cash”, “Momo”,..
PaymentMethod	Mô tả tên chi tiết hơn về phương thức thanh toán.

**2. Xác định các quan hệ thể hiện trong mô hình:****2.1. ĐĂNG NHẬP VÀO HỆ THỐNG (LOG-IN): USER – ACCOUNT**

- Mỗi người dùng (USER) được cấp một tài khoản (ACCOUNT) để đăng nhập vào hệ thống.
- Quan hệ này dùng để xác thực người dùng và cung cấp quyền truy cập vào các chức năng tương ứng của hệ thống.
- Mỗi tài khoản chỉ có thể cấp cho một người dùng duy nhất, nhưng không phải tất cả người dùng đều cần phải có tài khoản.

**2.2. LỰA CHỌN LIỆU TRÌNH (RECEIVES): CUSTOMER – SELECT\_TREATMENT**

- Khách hàng (CUSTOMER) có thể có các lựa chọn điều trị (SELECT\_TREATMENT).

- Một khách hàng có thể có nhiều lựa chọn điều trị, nhưng một lựa chọn điều trị chỉ có thể được áp dụng cho một khách hàng duy nhất. Tuy nhiên, không phải tất cả khách hàng đều cần có lựa chọn điều trị.

### **2.3. GẶP VẤN ĐỀ RĂNG MIỆNG (HAS): CUSTOMER – TOOTH\_PROBLEM**

- Mỗi khách hàng (CUSTOMER) gặp những vấn đề về răng miệng riêng (TOOTH\_PROBLEM).
- Một khách hàng có thể có một hoặc nhiều vấn đề về răng miệng, nhưng mỗi vấn đề về răng miệng chỉ tương ứng với một khách hàng duy nhất.

### **2.4. CHỐNG CHỈ ĐỊNH THUỐC (IS CONTRAINDICATED FOR): MEDICINE – CUSTOMER**

- Quan hệ này thể hiện sự chống chỉ định đối với một số loại thuốc (MEDICINE) của những khách hàng (CUSTOMER) có tình trạng sức khỏe đặc biệt.
- Một khách hàng có thể được chống chỉ định đối với một hoặc nhiều loại thuốc và một sự chống chỉ định đối với một loại thuốc cũng có thể liên kết tới nhiều khách hàng. Tuy nhiên, không phải tất cả khách hàng đều có sự chống chỉ định đối với một loại thuốc nào đó, và không phải tất cả loại thuốc đều có sự chống chỉ định đối với khách hàng.

### **2.5. ĐẶT LỊCH HẸN (TAKES): CUSTOMER – APPOINTMENT**

- Khách hàng (CUSTOMER) có thể đặt lịch hẹn khám (APPOINTMENT) với nha sĩ.
- Một khách hàng có thể đặt một hoặc nhiều cuộc hẹn với nha sĩ, nhưng một cuộc hẹn chỉ liên kết tới một khách hàng duy nhất. Tuy nhiên, không phải tất cả khách hàng đều đặt lịch hẹn với nha sĩ, trong khi tất cả các cuộc hẹn phải có khách hàng đặt hẹn.

### **2.6. ĐƯỢC HẸN TẠI PHÒNG (IS IN): APPOINTMENT – ROOM**

- Mỗi cuộc hẹn của khách hàng (APPOINTMENT) sẽ được sắp xếp diễn ra ở một phòng (ROOM) nào đó trong phòng khám nha khoa.
- Một cuộc hẹn chỉ diễn ra tại một phòng, trong khi một phòng có thể diễn ra nhiều cuộc hẹn ở các thời điểm khác nhau. Tuy nhiên, không phải tất cả các phòng đều có cuộc hẹn diễn ra, trong khi tất cả các cuộc hẹn phải được sắp xếp ở một phòng nào đó.

### **2.7. LẬP HOÁ ĐƠN (GENERATES): SELECT\_TREATMENT – INVOICE**

- Mỗi lựa chọn liệu trình điều trị (SELECT\_TREATMENT) sẽ tạo ra hóa đơn tương ứng (INVOICE).
- Một lựa chọn điều trị chỉ tạo ra một hóa đơn tương ứng, và một hóa đơn cũng chỉ tương ứng với một lựa chọn liệu trình điều trị duy nhất.

### **2.8. LÀ BÁC SĨ ĐIỀU TRỊ CHÍNH (IS MAIN DOCTOR): DENTIST – SELECT\_TREATMENT**

- Một nha sĩ (DENTIST) sẽ đảm nhận việc điều trị cho các lựa chọn điều trị (SELECT\_TREATMENT) khác nhau.
- Một nha sĩ có thể có một hoặc nhiều lựa chọn điều trị, nhưng một lựa chọn điều trị chỉ có thể được thực hiện bởi một nha sĩ duy nhất.

### **2.9. LÀ BÁC SĨ THỰC HIỆN CUỘC HẸN (IS A DOCTOR): DENTIST – APPOINTMENT**

- Nha sĩ (DENTIST) có thể có những cuộc hẹn (APPOINTMENT) với bệnh nhân của mình.
- Một nha sĩ có thể có một hoặc nhiều cuộc hẹn, nhưng một cuộc hẹn chỉ có thể được thực hiện bởi một nha sĩ duy nhất.

**2.10. ĐƯỢC THANH TOÁN BẰNG (IS PAID BY): INVOICE – PAYMENT\_METHOD**

- Hóa đơn (INVOICE) có thể được thanh toán bằng nhiều phương thức thanh toán (PAYMENT\_METHOD) khác nhau.
- Một hóa đơn chỉ có thể được thanh toán bằng một phương thức thanh toán duy nhất, nhưng một phương thức thanh toán có thể áp dụng trên nhiều hóa đơn. Tuy nhiên, không phải tất cả phương thức thanh toán đều được áp dụng trên các hóa đơn.

**2.11. ĐƯỢC CHỈ ĐỊNH BỞI (IS PRESCRIBED): MEDICINE – SELECT\_TREATMENT**

- Các loại thuốc (MEDICINE) được chỉ định dùng cho những lựa chọn điều trị khác nhau (SELECT\_TREATMENT).
- Một loại thuốc có thể được chỉ định cho nhiều lựa chọn điều trị, và một lựa chọn điều trị cũng có thể liên quan đến nhiều loại thuốc. Tuy nhiên, không phải tất cả các loại thuốc đều phải có trong các lựa chọn điều trị và không phải tất cả lựa chọn điều trị đều phải sử dụng thuốc.
- Mối liên hệ này bao gồm ba thuộc tính: Quantity (Số lượng thuốc của mỗi loại thuốc), UnitPrice (Giá tiền mỗi loại thuốc) và TotalPrice (Tổng giá tiền thuốc).

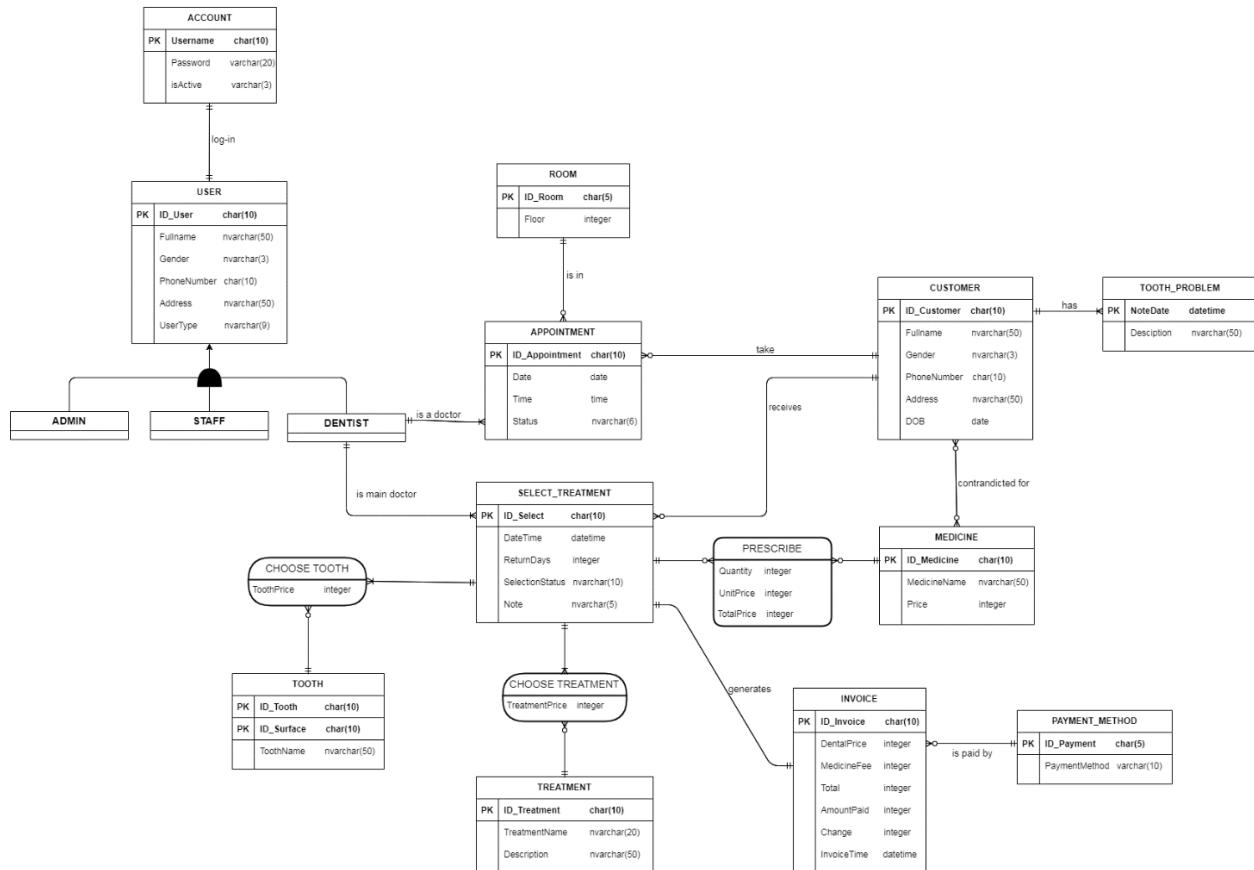
**2.12. CHỌN LIỆU TRÌNH ĐIỀU TRỊ (CHOOSE TREATMENT): SELECT\_TREATMENT – TREATMENT**

- Lựa chọn điều trị (SELECT\_TREATMENT) sẽ phát sinh ra liệu trình điều trị (TREATMENT).
- Một lựa chọn điều trị có thể có một hoặc nhiều liệu trình, và một liệu trình có thể liên kết tới một hoặc nhiều lựa chọn điều trị. Tuy nhiên, không phải tất cả liệu trình điều trị đều liên kết tới lựa chọn điều trị.
- Quan hệ này có một thuộc tính là Price (Chi phí điều trị).

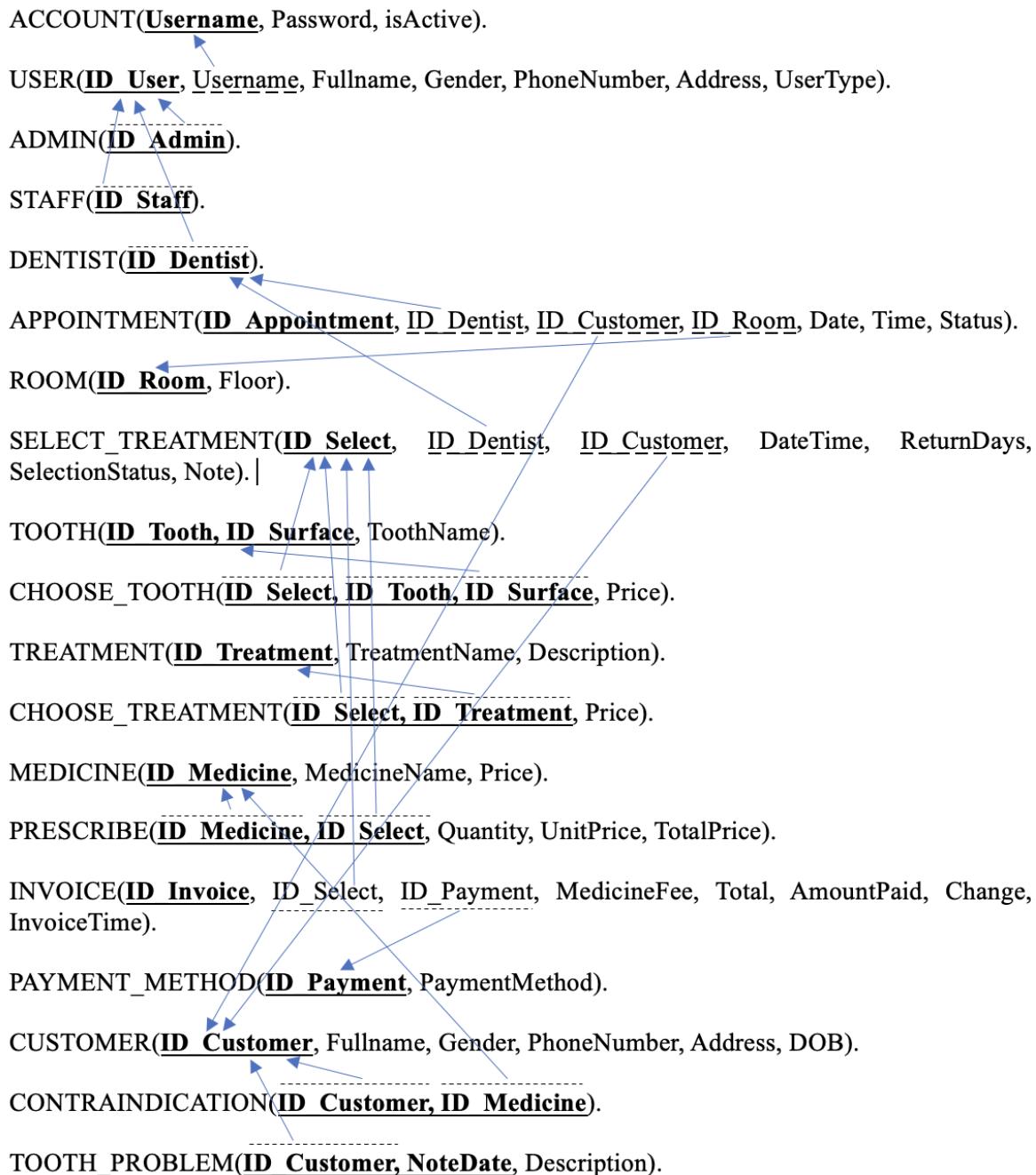
**2.13. CHỌN RĂNG (CHOOSE TOOTH): SELECT\_TREATMENT – TOOTH**

- Một lựa chọn điều trị (SELECT\_TREATMENT) sẽ được thực hiện trên các răng (TOOTH) đang gặp vấn đề.
- Một lựa chọn điều trị có thể được thực hiện trên một hoặc nhiều răng và một răng có thể được điều trị với một hoặc nhiều liệu trình. Tuy nhiên, không phải răng nào cũng cần được điều trị.
- Quan hệ này có một thuộc tính là Price (Chi phí điều trị răng).

### 3. Mô hình dữ liệu mức quan niệm:



#### 4. Lược đồ quan hệ:



#### 5. Phân tích phụ thuộc hàm - dạng chuẩn:

ACCOUNT:

- Username -> Password, isActive.

USER:

- ID User -> Username, Fullname, Gender, PhoneNumber, Address, UserType.

CUSTOMER:

- **ID\_Customer** -> Fullname, Gender, PhoneNumber, Address, DOB.

APPOINTMENT:

- **ID\_Appointment** -> ID\_Dentist, ID\_Customer, ID\_Room, Date, Time, Status.

ROOM:

- **ID\_Room** -> Floor.

TOOTH:

- **ID\_Tooth, ID\_Surface** -> ToothName.

TOOTH\_PROBLEM

- **ID\_Customer, NoteDate** -> Description.

MEDICINE:

- **ID\_Medicine** -> MedicineName, Price.

PREScribe:

- **ID\_Medicine, ID\_Select** -> Quantity, UnitPrice, TotalPrice.

TREATMENT:

- **ID\_Treatment** -> TreatmentName, Description.

SELECT\_TREATMENT:

- **ID\_Select** -> ID\_Dentist, ID\_Customer, DateTime, ReturnDays, SelectionStatus, Note.

CHOOSE\_TOOTH:

- **ID\_Select, ID\_Tooth, ID\_Surface** -> ToothPrice.

CHOOSE\_TREATMENT:

- **ID\_Select, ID\_Treatment** -> TreatmentPrice.

INVOICE:

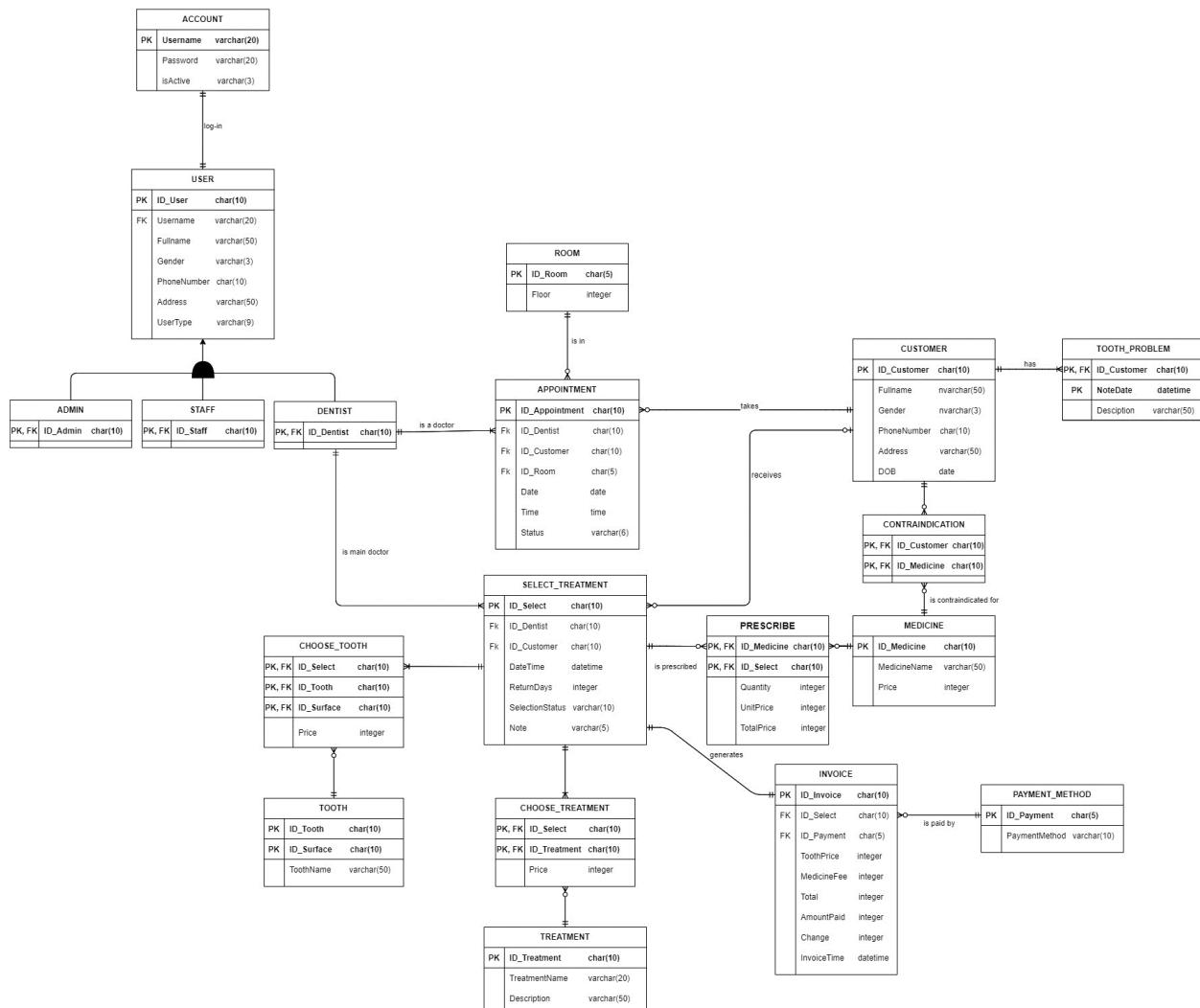
- **ID\_Invoice** -> ID\_Select, ID\_Payment, MedicineFee, Total, AmoutPaid, Change, InvoiceTime.

PAYMENT\_MODE:

- **ID\_Payment** -> PaymentMethod.

Lược đồ cơ sở dữ liệu đạt BCNF vì tất cả lược đồ con đạt BCNF.

## 6. Mô hình dữ liệu mức logic:



## VI. THIẾT KẾ CƠ SỞ DỮ LIỆU MỨC VẬT LÝ

### 1. Biên dịch thiết kế cơ sở dữ liệu ở mức vật lý:

#### 1.1. Thiết kế bảng cơ sở:

##### i. Bảng ACCOUNT:

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
Username	VARCHAR(20)	NOT NULL	Primary key
Password	VARCHAR(20)	NOT NULL	
isActive	VARCHAR(3)	“Yes” hoặc “No”	

- Tạo bảng:

```
-- ACCOUNT --
create table ACCOUNT
```

```

(
    Username varchar(20) not null,
    Pass_word varchar(20) not null,
    isActive varchar(3) not null default 'Yes',

    constraint PK_ACCOUNT
    primary key (Username),

    constraint CK_ACCOUNT_isActive
    check (isActive = 'Yes' or isActive = 'No'),
);

```

ii. Bảng USER\_DENTAL:

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_User	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
Username	VARCHAR(20)	NOT NULL	Foreign key → ACCOUNT(Username)
Fullname	VARCHAR(50)	NOT NULL	
Gender	VARCHAR(3)	“Nam” hoặc “Nữ”	
PhoneNumber	CHAR(10)	NOT NULL	
CurrAddress	VARCHAR(50)	NOT NULL	
UserType	VARCHAR(9)	“Admin”, “Dentist” hoặc “Staff”	

- Tạo bảng:

```

-- USER --
create table USER_DENTAL
(
    ID_User integer identity(1,1) not null,
    ID as 'U' + right('00000000' + cast(ID_User as varchar(10)),9)
persisted,
    Username varchar(20) not null,
    Fullname varchar(50) not null,
    Gender varchar(3) not null,
    PhoneNumber char(10) not null,
    CurrAddress varchar(50) not null,
    UserType varchar(9) not null,

    constraint PK_USERDENTAL
    primary key (ID_User),

    constraint CK_USER_Gender
    check (Gender = 'Male' or Gender = 'Female'),
);

```

```

constraint CK_USER_UserType
check (UserType = 'Staff' or UserType = 'Dentist' or UserType =
'Admin'),
);

alter table USER_DENTAL
add
constraint FK_USER_ACC
foreign key (Username)
references ACCOUNT(Username),

constraint UC_USER_Username
unique (Username)

```

iii. Bảng AD:

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Admin	INTEGER	NOT NULL	Primary key
			Foreign key → USER_DENTAL(ID_User)

- Tạo bảng:

```

-- ADMIN --
create table AD
(
    ID_Admin integer not null

    constraint PK_AD
    primary key (ID_Admin)
);

alter table AD
add
constraint FK_AD_USER
foreign key (ID_Admin)
references USER_DENTAL(ID_User)

```

iv. Bảng STAFF:

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Staff	INTEGER	NOT NULL	Primary key
			Foreign key → USER_DENTAL(ID_User)

- Tạo bảng:

```

-- STAFF --
create table STAFF
(

```

```

ID_Staff integer not null

constraint PK_STAFF
primary key (ID_Staff)
);

alter table STAFF
add
constraint FK_STAFF_USER
foreign key (ID_Staff)
references USER_DENTAL(ID_User)

```

v. Bảng DENTIST:

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Dentist	INTEGER	NOT NULL	Primary key Foreign key → USER_DENTAL(ID_User)

- Tạo bảng:

```

-- DENTIST --
create table DENTIST
(
    ID_Dentist integer not null

    constraint PK_DENTIST
    primary key (ID_Dentist)
);

alter table DENTIST
add
constraint FK_DENTIST_USER
foreign key (ID_Dentist)
references USER_DENTAL(ID_User)

```

vi. Bảng CUSTOMER

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Customer	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
Fullname	VARCHAR(50)	NOT NULL	
Gender	VARCHAR(3)	“M” hoặc “F”	
PhoneNumber	CHAR(10)	NOT NULL	
CurrAddress	VARCHAR(50)	NOT NULL	
DOB	DATE	NOT NULL	

- Tạo bảng:

```
-- CUSTOMER --
create table CUSTOMER
(
    ID_Customer integer identity(1,1) not null,
    ID as 'CU' + right('00000000' + cast(ID_Customer as varchar(10)),8)
persisted,
    Fullname varchar(50) not null,
    Gender varchar(3) not null,
    PhoneNumber char(10) not null,
    CurrAddress varchar(50) not null,
    DOB date not null,

    constraint PK_CUSTOMER
    primary key (ID_Customer),

    constraint CK_CUSTOMER_Gender
    check (Gender = 'M' or Gender = 'F')
);
```

### vii. Bảng APPOINTMENT

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Appointment	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
ID_Dentist	VARCHAR(10)	NOT NULL	Foreign key → DENTIST(ID_Dentist)
ID_Customer	VARCHAR(10)	NOT NULL	Foreign key → CUSTOMER(ID_Customer)
ID_Room	CHAR(5)	NOT NULL	Foreign key → ROOM(ID_Room)
Date_Appt	DATE	NOT NULL	
Time_Appt	TIME	NOT NULL	
Status_Appt	VARCHAR(8)	“New” hoặc “Reassess”	

- Tạo bảng:

```
-- APPOINTMENT --
create table APPOINTMENT
(
    ID_Appointment integer identity(1,1) not null,
    ID as 'AP' + right('00000000' + cast(ID_Appointment as
varchar(10)),8) persisted,
    ID_Dentist integer not null,
    ID_Customer integer not null,
    ID_Room char(5) not null,
    Date_Appt date not null,
    Time_Appt time not null,
```

```

Status_Appt varchar(8) not null,
constraint PK_APPOINTMENT
primary key (ID_Appointment),
constraint CK_APPT_Status
check (Status_Appt = 'New' or Status_Appt = 'Reassess')
);

alter table APPOINTMENT
add
constraint FK_APPOINTMENT_DENTIST
foreign key (ID_Dentist)
references DENTIST (ID_Dentist),
constraint FK_APPOINTMENT_CUSTOMER
foreign key (ID_Customer)
references CUSTOMER (ID_Customer),
constraint FK_APPOINTMENT_ROOM
foreign key (ID_Room)
references ROOM (ID_Room)

```

### viii. Bảng ROOM

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Room	CHAR(5)	NOT NULL	Primary key
Floor	INTEGER	1, 2 hoặc 3	

- Tạo bảng:

```

-- ROOM --
create table ROOM
(
    ID_Room char(5) not null,
    Floorr integer not null,

    constraint PK_ROOM
    primary key (ID_Room),

    constraint CK_ROOM_Floorr
    check (1 <= Floorr and Floorr <= 3)
);

```

### ix. Bảng TOOTH

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Tooth	INTEGER	NOT NULL	Primary key
ID_Surface	VARCHAR(10)	NOT NULL	Primary key

ToothName	VARCHAR(50)	NOT NULL	
-----------	-------------	----------	--

- Tạo bảng:

```
-- TOOTH --
create table TOOTH
(
    ID_Tooth integer identity(1,1) not null,
    ID_Surface varchar(10) not null,
    ToothName varchar(50) not null,

    constraint PK_TOOTH
    primary key (ID_Tooth, ID_Surface)
);
```

#### x. Bảng TOOTH\_PROBLEM

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
CustomerID	VARCHAR(10)	NOT NULL	Primary key
Decript	VARCHAR(50)	NOT NULL	
NotedDate	Datetime	NOT NULL	Primary key

- Tạo bảng:

```
-- TOOTH PROBLEM --
create table TOOTH_PROBLEM
(
    ID_Customer integer not null,
    NoteDate datetime not null default getdate(),
    Descript varchar(50) not null,

    constraint PK_TOOTHPROBLEM
    primary key (ID_Customer, NoteDate)
);

alter table TOOTH_PROBLEM
add
    constraint FK_TP_CUSTOMER
    foreign key (ID_Customer)
    references CUSTOMER (ID_Customer)
```

#### xi. Bảng MEDICINE

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Medicine	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
MedicineName	VARCHAR(50)	NOT NULL	
Price	INTEGER	NOT NULL	

- Tạo bảng:

```
-- MEDICINE --
create table MEDICINE
(
    ID_Medicine integer identity(1,1) not null,
    ID as 'ME' + right('00000000' + cast(ID_Medicine as varchar(10)),8)
persisted,
    MedicineName varchar(50) not null,
    Price integer not null,

    constraint PK_MEDICINE
    primary key (ID_Medicine)
);
```

### xii. Bảng CONTRAINDICATION

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Customer	INTEGER	NOT NULL	Primary key Foreign key → CUSTOMER(ID_Customer)
ID_Medicine	INTEGER	NOT NULL	
ID_Customer	INTEGER	NOT NULL	Primarykey Foreign key → MEDICINE(ID_Medicine)

- Tạo bảng:

```
-- CONTRAINDICATION --
create table CONTRAINDICATION
(
    ID_Customer integer not null,
    ID_Medicine integer not null,

    constraint PK_CONTRAINDICATION
    primary key (ID_Customer, ID_Medicine)
);

alter table CONTRAINDICATION
add
    constraint FK_CONTRAINDICATION_CUSTOMER
    foreign key (ID_Customer)
    references CUSTOMER (ID_Customer),

    constraint FK_CONTRAINDICATION_MEDICINE
    foreign key (ID_Medicine)
    references MEDICINE (ID_Medicine)
```

### xiii. Bảng TREATMENT

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Treatment	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
TreatmentName	VARCHAR(20)	NOT NULL	
Description	VARCHAR(50)	NOT NULL	

```
-- TREATMENT --
create table TREATMENT
(
    ID_Treatment integer identity(1,1) not null,
    ID as 'TR' + right('00000000' + cast(ID_Treatment as
varchar(10)),8) persisted,
    TreatmentName varchar(20) not null,
    Descript varchar(50) not null,

    constraint PK_TREATMENT
    primary key (ID_Treatment)
);
```

#### xiv. Bảng SELECT\_TREATMENT

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Select	INTEGER	NOT NULL	Primary key
ID	VARCHAR(10)	NOT NULL	
ID_Dentist	VARCHAR(10)	NOT NULL	Foreign key → DENTIST(ID_Dentist)
ID_Customer	VARCHAR(10)	NOT NULL	Foreign key → CUSTOMER(ID_Customer)
DateSelect	DATETIME	NOT NULL	
ReturnDays	INTEGER	NOT NULL	
SelectionStatus	VARCHAR(13)	“Planning”, “Completed” hoặc “Canceled”	
Note	VARCHAR(50)	NOT NULL	

- Tạo bảng:

```
-- SELECT_TREATMENT --
create table SELECT_TREATMENT
(
    ID_Select integer identity(1,1) not null,
    ID as 'ST' + right('00000000' + cast(ID_Select as varchar(10)),8)
persisted,
    ID_Dentist integer not null,
    ID_Customer integer not null,
    DateSelect datetime not null,
    ReturnDays integer,
```

```

SelectionStatus varchar(13) not null,
Note varchar(50),

constraint PK_SELECTTREATMENT
primary key (ID_Select),

constraint CK_ST_SelectionStatus
check (SelectionStatus = 'Planning' or SelectionStatus = 'Completed'
       or SelectionStatus = 'Canceled')
);

alter table SELECT_TREATMENT
add
constraint FK_ST_DENTIST
foreign key (ID_Dentist)
references DENTIST (ID_Dentist),

constraint FK_ST_CUSTOMER
foreign key (ID_Customer)
references CUSTOMER (ID_Customer)

```

xv. Bảng CHOOSE\_TOOTH

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Select	INTEGER	NOT NULL	Primary key Foreign key → SELECT_TREATMENT (ID_Select)
ID_Tooth	INTEGER	NOT NULL	
ID_Surface	VARCHAR(10)	NOT NULL	Primary key Foreign key → TOOTH(ID_Tooth)
Price	INTEGER	NOT NULL	

- Tạo bảng:

```

-- CHOOSE_TOOTH --
create table CHOOSE_TOOTH
(
    ID_Select integer not null,
    ID_Tooth integer not null,
    ID_Surface varchar(10) not null,
    Price integer not null,

    constraint PK_CHOOSETOOTH
    primary key (ID_Select, ID_Tooth, ID_Surface)
);

```

```

alter table CHOOSE_TOOTH
add
    constraint FK_CTOOTH_ST
    foreign key (ID_Select)
    references SELECT_TREATMENT (ID_Select),
    constraint FK_CTOOTH_TOOTH
    foreign key (ID_Tooth, ID_Surface)
    references TOOTH (ID_Tooth, ID_Surface)

```

xvi. Bảng CHOOSE\_TREATMENT

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Select	INTEGER	NOT NULL	Primary key Foreign key → SELECT_TREATMENT (ID_Select)
ID_Treatment	INTEGER	NOT NULL	
Price	INTEGER	NOT NULL	Primary key Foreign key → TREATMENT (ID_Treatment)

- Tạo bảng:

```

-- CHOOSE_TREATMENT --
create table CHOOSE_TREATMENT
(
    ID_Select integer not null,
    ID_Treatment integer not null,
    Price integer not null,

    constraint PK_CHOOSETREATMENT
    primary key (ID_Select, ID_Treatment)
);

alter table CHOOSE_TREATMENT
add
    constraint FK_CTREATMENT_STREATMENT
    foreign key (ID_Select)
    references SELECT_TREATMENT (ID_Select),

    constraint FK_CTREATMENT_TREATMENT
    foreign key (ID_Treatment)
    references TREATMENT (ID_Treatment)

```

xvii. Bảng INVOICE

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Invoice	INTEGER	NOT NULL	Primary key
ID	CHAR(10)	NOT NULL	
ID_Select	INTEGER	NOT NULL	Foreign key → SELECT_TRAETMENT (ID_Select)
ID_Payment	INTEGER	NOT NULL	Foreign key → PAYMENT_METHOD (ID_Payment)
DentalPrice	INTEGER		
MedicineFee	INTEGER		
Total	INTEGER		
AmountPaid	INTEGER	NOT NULL	
Change	INTEGER		
InvoiceTime	DATETIME	NOT NULL	

- Tạo bảng:

```
-- INVOICE --
create table INVOICE
(
    ID_Invoice integer identity(1,1) not null,
    ID as 'U' + right('00000000' + cast(ID_Invoice as varchar(10)),9)
persisted,
    ID_Select integer not null,
    ID_Payment integer not null,
    ToothPrice integer not null default 0,
    MedicineFee integer not null default 0,
    Total integer not null default 0,
    AmountPaid integer not null default 0,
    Changee integer not null default 0,
    InvoiceTime datetime not null default getdate(),

    constraint PK_INVOICE
    primary key (ID_Invoice)
);

alter table INVOICE
add
    constraint FK_INVOICE_STREATMENT
    foreign key (ID_Select)
    references SELECT_TREATMENT (ID_Select),

    constraint FK_INVOICE_PAYMENT
    foreign key (ID_Payment)
    references PAYMENT_METHOD (ID_Payment)
```

## xviii. Bảng PAYMENT\_METHOD

Thuộc tính	Kiểu dữ liệu	Ghi chú	Khóa
ID_Payment	INTEGER	NOT NULL	Primary key
ID	CHAR(5)	NOT NULL	
PaymentMethod	VARCHAR(10)	“Cash”, “Momo”,...	

- Tạo bảng:

```
-- PAYMENT_METHOD --
create table PAYMENT_METHOD
(
    ID_Payment integer identity(1,1) not null,
    ID as 'PM' + right('000' + cast(ID_Payment as varchar(5)),3)
persisted,
    PaymentMethod varchar(10) not null,
    constraint PK_PAYEMENT_METHOD
    primary key (ID_Payment)
);
```

## 1.2. Tạo trigger quản lý dữ liệu khi nhập vào bảng:

- Bảng PRESCRIBE:
- UnitPrice của mỗi loại thuốc được lấy từ Price của bảng MEDICINE.
- Tổng tiền mỗi loại thuốc được kê đơn (TotalPrice) được tính bằng cách nhân số lượng (Quantity) với đơn giá (UnitPrice) của loại thuốc đó.

```
create trigger tg_add_TotalPrice_PRESCRIBE on PRESCRIBE
after insert
as
begin
    declare @id_select int
    declare @id_medicine int
    declare @unitPrice int
    declare @quantity int

    select @id_select = ID_Select,
           @id_medicine = ID_Medicine,
           @quantity = Quantity
    from inserted

    -- UnitPrice(PRESCRIBE) = Price(MEDICINE)
    select @unitPrice = Price
    from MEDICINE where ID_Medicine = @id_medicine

    update PRESCRIBE
    set UnitPrice = @unitPrice
    where ID_Select = @id_select and ID_Medicine = @id_medicine
```

```
-- TotalPrice(PRESCRIBE) = Quantity * UnitPrice
update PRESCRIBE
set TotalPrice = @unitPrice * @quantity
where ID_Select = @id_select and ID_Medicine = @id_medicine
end
```

- Bảng INVOICE:

- ToothPrice của bảng INVOICE được tính bằng tổng giá tiền của tất cả các răng (Price(CHOOSE\_TOOTH)) và phương pháp điều trị (Price(CHOOSE\_TREATMENT)) được chọn.
- MedicinePrice của bảng INVOICE được tính bằng tổng tiền các loại thuốc được kê đơn (TotalPrice(PRESCRIBE))
- Tổng tiền hóa đơn (Total) được tính bằng tổng tiền răng (ToothPrice) và tiền thuốc kê đơn (MedicinePrice).
- Nếu khách hàng chọn thanh toán bằng thẻ thì tiền thối lại sẽ bằng 0 (Change = 0), nếu khách thanh toán bằng tiền mặt thì tiền thối sẽ bằng tiền khách đưa trừ cho tổng hóa đơn (Change = AmountPaid - Total)

```
create trigger tg_add_Total_INVOICE on INVOICE
after insert
as
begin
    declare @id_invoice int
    declare @id_select int
    declare @id_payment int
    declare @totalToothPrice int
    declare @totalMedicinePrice int
    declare @total int

    select @id_invoice = ID_Invoice,
           @id_select = ID_Select,
           @id_payment = ID_Payment
    from inserted

    -- ToothPrice(INVOICE) = Price(CHOOSE_TOOTH) +
    Price(CHOOSE_TREATMENT)
    declare @treatmentPrice int
    declare @toothPrice int

    select @treatmentPrice = ISNULL(SUM(Price), 0)
    from CHOOSE_TREATMENT where ID_Select = @id_select

    select @toothPrice = ISNULL(SUM(Price), 0)
    from CHOOSE_TOOTH where ID_Select = @id_select

    set @totalToothPrice = @treatmentPrice + @toothPrice
```

```

update INVOICE
set ToothPrice = @totalToothPrice
where ID_Invoice = @id_invoice

-- MedicineFee(INVOICE) = TotalPrice(PRESCRIBE)
select @totalMedicinePrice = ISNULL(SUM(TotalPrice), 0)
from PRESCRIBE where ID_Select = @id_select

update INVOICE
set MedicineFee = @totalMedicinePrice
where ID_Invoice = @id_invoice

-- Total(INVOICE) = ToothPrice + MedicineFee
set @total = @totalToothPrice + @totalMedicinePrice

update INVOICE
set Total = @total
where ID_Invoice = @id_invoice

-- Change(INVOICE) = AmountPaid - Total (ID_Payment = 'Cash')
declare @isByCash char(4)
set @isByCash = 'Cash'

if @id_payment = @isByCash
begin
    update INVOICE
    set Changee = AmountPaid - @total
    where ID_Invoice = @id_invoice
end
-- Change(INVOICE) = 0 (ID_Payment != 'Cash')
-- AmountPaid(INVOICE) = Total
else
begin
    update INVOICE
    set Changee = 0
    where ID_Invoice = @id_invoice

    update INVOICE
    set AmountPaid = @total
    where ID_Invoice = @id_invoice
end
end

```

### 1.3. Thiết kế business rules:

- Bảng APPOINTMENT:
  - Trong trường hợp thêm hoặc cập nhật lịch hẹn, kiểm tra xem nha sĩ có bị trùng lịch hẹn hay không.

```
create trigger tg_check_ApptDentist_APPOINTMENT on APPOINTMENT
```

```

after insert, update
as
begin
    if exists (select *
                from APPOINTMENT a1, APPOINTMENT a2
                where a1.ID_Appointment != a2.ID_Appointment
                and a1.Date_Appt = a2.Date_Appt
                and a1.Time_Appt = a2.Time_Appt
                and a1.ID_Dentist = a2.ID_Dentist
            )
        begin
            rollback
        end
    end

```

## 2. Nối các phần tiến trình thực hiện câu lệnh với các bảng:

STT	Giao tác phổ biến	Bảng truy vấn liên quan
1	Thêm mới, xem, cập nhật và xoá thông tin cá nhân của người dùng	<ul style="list-style-type: none"> <li>- ACCOUNT</li> <li>- USER: FOREIGN KEY (Username) REFERENCES ACCOUNT(Username)</li> <li>- ADMIN: FOREIGN KEY (ID_Admin) REFERENCES USER(ID_User)</li> <li>- STAFF: FOREIGN KEY (ID_Staff) REFERENCES USER(ID_User)</li> <li>- DENTIST: FOREIGN KEY (ID_Dentist) REFERENCES USER(ID_User)</li> <li>- APPOINTMENT: FOREIGN KEY (ID_Dentist) REFERENCES DENTIST(ID_Dentist)</li> <li>- SELECT_TREATMENT: FOREIGN KEY (ID_Dentist) REFERENCES DENTIST(ID_Dentist)</li> </ul>
2	Xem, cập nhật thông tin tài khoản của người dùng	ACCOUNT
3	Thêm khách hàng mới, xem và cập nhật thông tin cá nhân của khách hàng	CUSTOMER
4	Xem và cập nhật liệu trình điều tri	TREATMENT
5	Thêm mới, xem và cập nhật thông tin thuốc	MEDICINE

STT	Giao tác phổ biến	Bảng truy vấn liên quan
6	Xem thông tin răng	<b>TOOTH</b>
7	Xem phương thức thanh toán	<b>PAYMENT_METHOD</b>
8	Xem danh sách các phòng tại phòng khám	<b>FLOOR</b>
9	Thêm mới, xem, cập nhật, xoá thông tin răng đang gặp vấn đề	<ul style="list-style-type: none"> <li>- <b>CUSTOMER</b></li> <li>- <b>TOOTH_PROBLEM:</b> FOREIGN KEY (ID_Customer) REFERENCES CUSTOMER(ID_Customer)</li> </ul>
10	Thêm mới, xem, cập nhật hóa đơn	<ul style="list-style-type: none"> <li>- <b>INVOICE:</b> + FOREIGN KEY (ID_Select) REFERENCES SELECT_TREATMENT(ID_Select)</li> <li>+ FOREIGN KEY (ID_Payment) REFERENCES PAYMENT_METHOD(ID_Payment)</li> <li>- <b>SELECT_TREATMENT</b></li> <li>- <b>PAYMENT_METHOD</b></li> </ul>
11	Thêm mới, xem, cập nhật, xoá lịch hẹn	<ul style="list-style-type: none"> <li>- <b>APPOINTMENT:</b> + FOREIGN KEY (ID_Dentist) REFERENCES DENTIST(ID_Select)</li> <li>+ FOREIGN KEY (ID_Customer) REFERENCES CUSTOMER(ID_Customer)</li> <li>+ FOREIGN KEY (ID_Room) REFERENCES ROOM(ID_Room)</li> <li>- <b>DENTIST</b></li> <li>- <b>CUSTOMER</b></li> <li>- <b>ROOM</b></li> </ul>
12	Thêm mới, xem, cập nhật, xoá chống chỉ định thuốc	<ul style="list-style-type: none"> <li>- <b>CONTRAINICATION:</b> + FOREIGN KEY (ID_Customer) REFERENCES CUSTOMER(ID_Customer)</li> <li>+ FOREIGN KEY (ID_Medicine) REFERENCES MEDICINE(ID_Medicine)</li> <li>- <b>MEDICINE</b></li> <li>- <b>CUSTOMER</b></li> </ul>
13	Thêm mới, xem, cập nhật chỉ định thuốc của nha sĩ	<ul style="list-style-type: none"> <li>- <b>PREScribe:</b> + FOREIGN KEY (ID_Medicine) REFERENCES MEDICINE(ID_Medicine)</li> <li>+ FOREIGN KEY (ID_Customer) REFERENCES SELECT_TREATMENT(ID_Select)</li> <li>- <b>MEDICINE</b></li> <li>- <b>SELECT_TREATMENT</b></li> </ul>
14	Thêm mới, xem, cập nhật, xoá lựa chọn điều trị	<ul style="list-style-type: none"> <li>- <b>SELECT_TREATMENT:</b> + FOREIGN KEY (ID_Dentist) REFERENCES DENTIST(ID_Dentist)</li> </ul>

STT	Giao tác phổ biến	Bảng truy vấn liên quan
		FOREIGN KEY (ID_Customer) REFERENCES CUSTOMER(ID_Customer) <ul style="list-style-type: none"> <li>- DENTIST</li> <li>- CUSTOMER</li> </ul>
15	Thêm mới, xem, cập nhật, xoá lựa chọn răng cần điều trị	<ul style="list-style-type: none"> <li>- CHOOSE_TOOTH:</li> <li>+ FOREIGN KEY (ID_Select) REFERENCES SELECT_TREATMENT(ID_Select)</li> <li>+ FOREIGN KEY (ID_Tooth, ID_Surface) REFERENCES TOOTH(ID_Tooth, ID_Surface)</li> <li>- SELECT_TREATMENT</li> <li>- TOOTH</li> </ul>
16	Thêm mới, xem, cập nhật, xoá lựa chọn liệu trình điều trị	<ul style="list-style-type: none"> <li>- CHOOSE_TREATMENT:</li> <li>+ FOREIGN KEY (ID_Select) REFERENCES SELECT_TREATMENT(ID_Select)</li> <li>+ FOREIGN KEY (ID_Select) REFERENCES TREATMENT(ID_Treatment)</li> <li>- SELECT_TREATMENT</li> <li>- TREATMENT</li> </ul>

### 3. Ánh xạ các tiến trình vào bảng:

TABLE	TRANSACTION	1		2		3		4		5		6		7		8					
		R	I	U	D	R	I	U	D	R	I	U	D	R	I	U	D	R	I	U	
ACCOUNT		x	x	x	x	x	x														
USER		x	x	x	x																
ADMIN		x		x																	
STAFF		x		x																	
DENTIST		x		x																	
SELECT_TREATMENT				x																	
APPOINTMENT				x																	
ROOM																		x			
TOOTH																		x			
TREATMENT								x	x												
CHOOSE_TOOTH																					
CHOOSE_TREATMENT																					
CUSTOMER							x	x	x												
TOOTH_PROBLEM																					
INVOICE																					
PAYMENT_METHOD												x	x	x				x			
MEDICINE																					
CONTRAINDICATION													x	x	x						
PRESCRIBE													x	x	x						
		R	I	U	D	R	I	U	D	R	I	U	D	R	I	U	D	R	I	U	D
ACCOUNT																					
USER																					
ADMIN																					
STAFF																					
DENTIST							x								x		x	x	x	x	x
SELECT_TREATMENT				x											x		x	x	x	x	x
APPOINTMENT					x	x	x	x	x												
ROOM					x																
TOOTH						x											x				
TREATMENT																		x			
CHOOSE_TOOTH																	x	x	x	x	x
CHOOSE_TREATMENT																		x	x	x	x
CUSTOMER	x						x			x							x				
TOOTH_PROBLEM	x	x	x	x	x					x							x				
INVOICE				x	x	x															
PAYMENT_METHOD				x																	
MEDICINE										x		x		x							
CONTRAINDICATION										x	x	x	x				x	x	x		
PRESCRIBE											x	x	x								

Nhận xét:

- Dựa trên tần suất truy vấn vào các bảng trong CSDL, ta nhận thấy rằng truy vấn SELECT (Đọc) được thực hiện nhiều nhất.
- Các bảng có nhiều truy vấn SELECT nhất gồm có:  
**CUSTOMER:** Lưu trữ thông tin cá nhân của khách hàng.

**SELECT\_TREATMENT:** Lưu trữ danh sách các lựa chọn điều trị của khách hàng.

**MEDICINE:** Lưu trữ danh sách thuốc trong kho thuốc.

**ROOM:** Lưu trữ danh sách các phòng trong phòng khám.

**TOOTH:** Lưu trữ thông tin các răng.

**TREATMENT:** Lưu trữ danh sách các liệu trình điều trị mà phòng khám có.

**PAYMENT\_METHOD:** Lưu trữ các phương thức thanh toán mà phòng khám sử dụng.

Và Khi tối ưu hoá CSDL, ta xem xét cài đặt chỉ mục (index) cho các trường quan trọng trong các bảng này để cải thiện hiệu suất truy vấn.

#### 4. Phi chuẩn hóa:

PRESCRIBE	
PK, FK	ID_Medicine char(10)
PK, FK	ID_Select char(10)
	Quantity integer
	UnitPrice integer
	TotalPrice integer

Bảng PRESCRIBE:

- Cột UnitPrice chứa đơn giá của từng loại thuốc được kê đơn cho việc điều trị.
- Việc lưu trữ đơn giá trực tiếp trong PRESCRIBE giúp cho ta có thể xem thông tin chi tiết đơn thuốc mà không cần phải thông qua bảng MEDICINE, điều này giảm tải cho hệ thống và cải thiện tốc độ truy vấn.

## VII. CẢI THIỆN HIỆU QUẢ TRUY VẤN

### 1. Phân tích hiệu năng:

- Phân tích tần suất sử dụng:
  - Giả sử nha khoa bao gồm 10 phòng khám khác nhau. Mỗi phòng sẽ gồm 2 nha sĩ: một nha sĩ kê đơn thuốc và 1 nha sĩ khám bệnh.
  - Trong mỗi ngày, một phòng khám hoạt động từ 9h đến 18h và nghỉ trưa từ 12h đến 13h. Vậy mỗi ngày phòng khám làm việc 8 giờ.
  - Mỗi giờ trong các ngày sẽ có trung bình 10-15 lượt khách đến khám. Vậy một ngày sẽ có 80-120 lượt khách. Vào các ngày cuối tuần, thường sẽ có nhiều lượt khách hơn nên trung bình 1 giờ trong các ngày này sẽ có từ 20-30 lượt khách nên một ngày có trung bình 160-240 lượt khách.
  - Mỗi khách sẽ cần khám chọn 1-6 bờ mặt răng và trung bình một khách sẽ chọn 2 răng.
  - Số lần nha sĩ xem thông tin thuốc để kê đơn cho bệnh nhân trùng với số lượt khách mỗi ngày. Bỏ qua chênh lệch nhỏ trong tình huống do nha sĩ kê đơn nhầm.
  - Trong một đơn thuốc trung bình nha sĩ kê từ 2-3 loại thuốc.
  - Có 3 loại phương thức thanh toán.
  - Mỗi hóa đơn nhân viên sẽ xem trung bình 2-3 lần để kiểm tra lại thông tin và in hóa đơn.
  - Đối với thực thể SELECT\_TREATMENT: số lượt truy cập sẽ trùng với số lượt khách. Vậy số lượt truy cập trung bình mỗi tiếng là 10-15 lần và cao điểm sẽ là 20-30 lần.
  - Đối với thực thể CHOOSE\_TOOTH: số lượt truy cập sẽ là số lượt khách nhân với răng và loại răng. Vậy số lượt truy cập trung bình mỗi tiếng là 20-180 lần và cao điểm sẽ là 40-180 lần.
  - Đối với thực thể DETAIL\_MEDICINE: số lượt truy cập sẽ là số lượt nha sĩ kê đơn nhân với số lượt khách và số thuốc cần kê. Vậy số lượt truy cập trung bình mỗi giờ sẽ là 200-450 lần và cao điểm sẽ là 400-900 lần.
  - Đối với MODE\_PAYMENT: số lượt truy cập bằng với số lượt khách. Vậy số lượt truy cập trung bình mỗi tiếng là 10-15 lần và cao điểm là 20-30 lần.
  - Đối với INVOICE: số lượt truy cập bằng với số lượt khách nhân với số lần nhân viên xem. Vậy số lượt truy cập trung bình mỗi tiếng là 10-15 lần và cao điểm là 20-30 lần.

Thực thể	Loại truy cập	Số lần truy cập mỗi tiếng	
		Trung bình	Cao điểm
SELECT_TREATMENT	Đọc	10-15	20-30
CHOOSE_TOOTH	Đọc	10-60	20-120
DETAIL_MEDICINE	Đọc	200-450	400-900
MODE_PAYMENT	Đọc	10-15	20-30
INVOICE	Đọc	20-45	40-90

## 2. Thiết lập chỉ mục (index):

```
USE QLPKNHAKHOA
```

```
--Create index on column DOB of table CUSTOMER
```

```
CREATE NONCLUSTERED INDEX index_CUSTOMER_DOB
ON CUSTOMER (DOB)
```

```
GO
```

```
--Create index on column ID_Customer of table SELECT_TREATMENT
```

```
CREATE NONCLUSTERED INDEX index_SELECTTREATMENT_IDCustomer
ON SELECT_TREATMENT (ID_Customer)
```

```
GO
```

- Thiết lập chỉ mục trên thuộc tính DOB của quan hệ CUSTOMER và thuộc tính ID\_Customer của quan hệ SELECT\_TREATMENT.
- Sau khi cài đặt chỉ mục, tốc độ truy vấn được cải thiện đáng kể:
- Trên quan hệ SELECT\_TREATMENT:
  - + Trước khi cài đặt chỉ mục:

EventClass	TextData	ApplicationName	NTUserName	LoginName	ClientProcessID	SPID	StartTime	CPU	Reads	Writes	Duration	DatabaseID	DatabaseName	HostName
sql_batch_init	SET ANSI_WARNINGS ON	azdata		sa	32456	57	2023-12-1...	1000	0	0	973	1	master	Chis-MacBook-Pro
sql_batch_init	SELECT * FROM master..sysprocesses	azdata		sa	32456	57	2023-12-1...					1	master	Chis-MacBook-Pro
sql_batch_init	SELECT IS_MEMBER('sa')	azdata		sa	32456	57	2023-12-1...	0	0	0	942	1	master	Chis-MacBook-Pro
sql_batch_init	SELECT name FROM master..sysdatabases	azdata		sa	32456	57	2023-12-1...					1	master	Chis-MacBook-Pro
sql_batch_init	SELECT name FROM master..syslogins	azdata		sa	32456	57	2023-12-1...	17000	152	0	20765	1	master	Chis-MacBook-Pro
sql_batch_init	SET NOEXEC ON	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacBook-Pro
sql_batch_init	SET NOEXEC OFF	azdata		sa	32456	61	2023-12-1...	3000	0	0	3370	15	QLPKNHA...	Chis-MacBook-Pro
sql_batch_init	+!SELECT * FROM master..sysprocesses	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacBook-Pro
sql_batch_init	+!SELECT * FROM master..syslogins	azdata		sa	32456	61	2023-12-1...	466000	2543	2	414814	15	QLPKNHA...	Chis-MacBook-Pro
logout		azdata		sa	32456	65	2023-12-1...	37000	5479	11	229477000			Chis-MacBook-Pro
rpc_complaint	exec sp_repair_db	azdata		sa	32456	65	2023-12-1...	1000	0	0	790	15	QLPKNHA...	Chis-MacBook-Pro

Text      Details

```
1
2 SELECT * FROM SELECT_TREATMENT WHERE ID_Customer = 111
```

- + Sau khi cài đặt chỉ mục:

EventClass	TextData	Application	NTUserName	LoginName	ClientProcessID	SPID	StartTime	CPU	Reads	Writes	Duration	DatabaseID	DatabaseName	HostName
sql_batch... SET ANSI...	azdata		sa	32456	57	2023-12-1...	1000	0	0	890	1	master	Chis-MacB...	
sql_batch... SELECT IS...	azdata		sa	32456	57	2023-12-1...					1	master	Chis-MacB...	
sql_batch... SELECT IS...	azdata		sa	32456	57	2023-12-1...	0	0	0	893	1	master	Chis-MacB...	
sql_batch... SELECT na...	azdata		sa	32456	57	2023-12-1...					1	master	Chis-MacB...	
sql_batch... SELECT na...	azdata		sa	32456	57	2023-12-1...	10000	152	0	12536	1	master	Chis-MacB...	
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...	4000	0	0	4458	15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...	5000	14	0	5592	15	QLPKNHA...	Chis-MacB...	
logout	azdata		sa	32456	63	2023-12-1...	26000	88727	42	119690000			Chis-MacB...	
rpc_compli...	exec sp_re...	azdata		sa	32456	63	2023-12-1...	0	0	0	364	15	QLPKNHA...	Chis-MacB...
login	-- network ...	azdata		sa	32456	63	2023-12-1...				15		Chis-MacB...	
<b>Text</b>		<b>Details</b>												
1 SELECT * FROM SELECT_TREATMENT WHERE ID_Customer = 111														

- Trước khi cài đặt index, CPU time là 466000 ms. Sau khi cài đặt index, CPU time giảm còn 5000 ms. Tốc độ truy vấn được cải thiện lớn.
- Trên quan hệ CUSTOMER:
    - + Trước khi cài đặt chỉ mục:

EventClass	TextData	Application	NTUserName	LoginName	ClientProcessID	SPID	StartTime	CPU	Reads	Writes	Duration	DatabaseID	DatabaseName	HostName
sql_batch... SELECT na...	azdata		sa	32456	57	2023-12-1...					1	master	Chis-MacB...	
sql_batch... SELECT na...	azdata		sa	32456	57	2023-12-1...	22000	152	0	23675	1	master	Chis-MacB...	
logout	azdata		sa	32456	51	2023-12-1...	0	0	0	439900000			Chis-MacB...	
login	-- network ...	azdata		sa	32456	51	2023-12-1...				1		Chis-MacB...	
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...	4000	0	0	4145	15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...	609000	2617	2	517152	15	QLPKNHA...	Chis-MacB...	
sql_batch... DECLARE ...	azdata		sa	32456	55	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... DECLARE ...	azdata		sa	32456	55	2023-12-1...	11000	0	0	11329	15	QLPKNHA...	Chis-MacB...	
rpc_compli...	exec sp_ex...	azdata		sa	32456	55	2023-12-1...	6000	16	0	8540	15	QLPKNHA...	Chis-MacB...
<b>Text</b>		<b>Details</b>												
1 SELECT * FROM CUSTOMER WHERE YEAR(DOB) = '2003'														

- + Sau khi cài đặt chỉ mục:

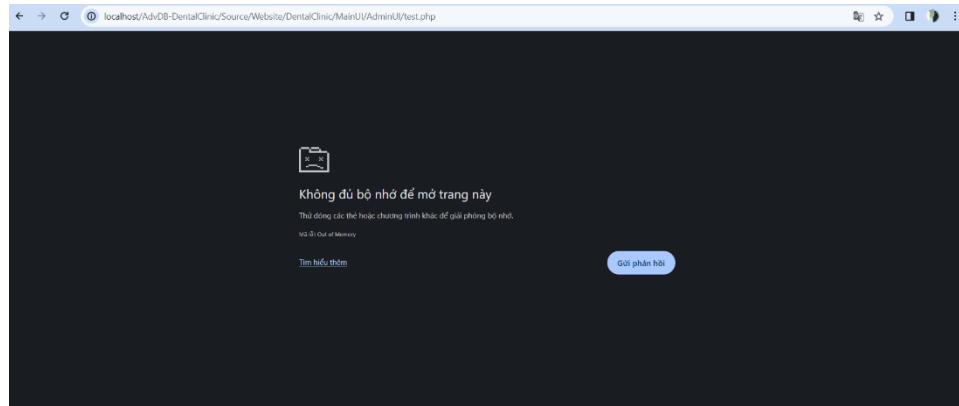
EventClass	TextData	Application	NTUserName	LoginName	ClientProcessID	SPID	StartTime	CPU	Reads	Writes	Duration	DatabaseID	DatabaseName	HostName
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SET NOEX...	azdata		sa	32456	61	2023-12-1...	6000	0	0	5051	15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... SELECT * F...	azdata		sa	32456	61	2023-12-1...	77000	1408	0	82649	15	QLPKNHA...	Chis-MacB...	
logout	azdata		sa	32456	68	2023-12-1...	12000	100	0	48207000			Chis-MacB...	
rpc_compli...	exec sp_re...	azdata		sa	32456	68	2023-12-1...	0	0	0	405	15	QLPKNHA...	Chis-MacB...
login	-- network ...	azdata		sa	32456	68	2023-12-1...				15		Chis-MacB...	
sql_batch... DECLARE ...	azdata		sa	32456	68	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
sql_batch... DECLARE ...	azdata		sa	32456	68	2023-12-1...	7000	0	0	8521	15	QLPKNHA...	Chis-MacB...	
rpc_compli...	exec sp_ex...	azdata		sa	32456	68	2023-12-1...	8000	16	0	9673	15	QLPKNHA...	Chis-MacB...
sql_batch... select SER...	azdata		sa	32456	68	2023-12-1...					15	QLPKNHA...	Chis-MacB...	
<b>Text</b>		<b>Details</b>												
1 SELECT * FROM CUSTOMER WHERE YEAR(DOB) = '2003'														

- Trước khi cài đặt index, CPU time là 609000 ms. Sau khi cài đặt index, CPU time giảm còn 77000 ms. Tốc độ truy vấn được cải thiện lớn.

### 3. Đề xuất tối ưu hóa truy vấn:

#### 3.1. Kỹ thuật phân trang:

- Khi truy vấn dữ liệu của một bảng có quá nhiều dòng dữ liệu sẽ dẫn đến việc quá tải và trang web sẽ không thể hiển thị được dữ liệu, ví dụ như trong cơ sở dữ liệu QLPKNHAKHOA, bảng CUSTOMER bao gồm xấp xỉ 10000 dòng dữ liệu, nếu thực hiện SELECT \* FROM CUSTOMER, trang web sẽ bị quá tải như hình bên dưới:



- Do đó, việc phân trang dữ liệu là rất cần thiết giúp vừa cải thiện hiệu quả truy suất dữ liệu từ cơ sở dữ liệu, vừa tránh trường hợp gây quá tải cho trang web. Theo đó, thay vì chỉ truy vấn toàn bộ dữ liệu trong một lần, ta thực hiện chia nhỏ 10000 dòng dữ liệu thành nhiều phần tương ứng với số dòng dữ liệu mong muốn cho từng trang.
- Hàm giúp phân trang thực hiện như sau:

```
function getAllWithPagination($tableName, $pageSize, $pageNumber, $pageOrder)
{
    global $conn;

    $table = validate($tableName);
    $startRow = ($pageNumber - 1) * $pageSize;

    $query = "SELECT COUNT(*) as total FROM $table"; // Đếm tổng số dòng
    $result = sqlsrv_query($conn, $query);
    $totalRows = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)[['total']];

    $query = "SELECT * FROM $table ORDER BY $pageOrder OFFSET $startRow ROWS FETCH NEXT $pageSize ROWS ONLY";
    $result = sqlsrv_query($conn, $query);

    if ($result) {
        $data = array();
        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
            $data[] = $row;
        }

        $response = [
            'status' => 'Data Found',
            'data' => $data,
            'total' => $totalRows,
        ];
    } else {
        $response = [
            'status' => 'Something went wrong! Please try again.',
        ];
    }

    return $response;
}
```

- Dữ liệu cần truyền vào bao gồm tên bảng cần truy vấn, số dòng dữ liệu mong muốn có trong một trang, số thứ tự của trang và cột dùng để sắp xếp thứ tự tăng dần. Ví dụ như ta cần truy vấn dữ liệu cho trang thứ 2 từ bảng SELECT\_TREATMENT với mỗi trang gồm 20 dòng dữ liệu và sắp xếp theo thứ tự tăng dần ID\_Select thì lời gọi hàm sẽ là getAllWithPagination('SELECT\_TREATMENT', 20, 2, 'ID\_Select').

- Bước đầu tiên hàm thực hiện sẽ tính toán xem dòng dữ liệu đầu tiên cần truy xuất. Trong SQL Server, khi cần truy xuất dữ liệu trong dòng thứ n thì cần bắt đầu từ OFFSET thứ n-1. Khi đó:
  - Dòng đầu tiên của trang 1 là dòng dữ liệu số 1, bắt đầu từ OFFSET  $0 = (1-1) * \text{pageSize}$ .
  - Dòng đầu tiên của trang 2 là dòng dữ liệu số  $1 + \text{pageSize}$ , bắt đầu từ OFFSET  $\text{pageSize} = (2-1) * \text{pageSize}$ .
  - Dòng đầu tiên của trang 3 là dòng dữ liệu số  $1 + 2 * \text{pageSize}$ , bắt đầu từ OFFSET  $2 * \text{pageSize} = (3-1) * \text{pageSize}$ .

→ Công thức tính dòng đầu tiên của trang dữ liệu là :

$$\text{Start Row} = (\text{Page Number} - 1) \times \text{Page Size}.$$

- Tiếp theo, hàm thực hiện đếm số dòng có trong bảng để có thể hiển thị thông tin phân trang lên trang web.
- Cuối cùng, hàm sẽ thực hiện nạp câu query :

```
"SELECT * FROM $table ORDER BY $pageOrder
OFFSET $startRow ROWS
FETCH NEXT $pageSize ROWS ONLY"
```

- Câu query thực hiện lấy toàn bộ số dòng từ bảng \$table với thứ tự tăng dần theo \$pageOrder, bỏ qua \$startRow dòng dữ liệu đầu tiên để lấy \$pageSize dòng kế tiếp đó.
- **Ví dụ:**

`SELECT * FROM CUSTOMER ORDER BY ID_Customer OFFSET 120 ROWS FETCH NEXT 20 ROWS ONLY` : lấy 20 dòng dữ liệu từ bảng CUSTOMER được sắp xếp theo thứ tự tăng dần ID\_CUSTOMER bỏ qua 120 dòng đầu tiên. Qua đó, câu truy vấn thực hiện lấy dữ liệu cho trang thứ 7.

### 3.2. Cải thiện câu Select:

- Tuy nhiên, câu truy vấn thực hiện trong đoạn code ở phần phân trang vẫn cần cải thiện bởi trong trường hợp bảng có quá nhiều cột dữ liệu, việc thực hiện `SELECT * FROM $table` sẽ dẫn đến việc truy xuất và lưu trữ dữ liệu không hợp lý và kém hiệu quả. Chính vì thế, tùy vào trường hợp cụ thể ta cần chọn ra những cột dữ liệu cần thiết để truy xuất, giảm thiểu sử dụng quá nhiều bộ nhớ để lưu trữ những giá trị không cần thiết. Ví dụ như trong khi chọn nha sĩ cho một cuộc hẹn, người dùng chỉ cần biết tên nha sĩ để chọn, thay vì chọn toàn bộ dữ liệu, ta chỉ chọn cột 'Fullname'.

`'SELECT * FROM DENTIST' → 'SELECT Fullname FROM DENTIST'.`

### 4. Các hàm được sử dụng để truy vấn dữ liệu trong ứng dụng:

- Hàm chèn dòng dữ liệu mới vào bảng với dữ liệu truyền vào là tên bảng và mảng kết hợp : “tên cột” => “dữ liệu”. Hàm thực hiện lấy ra tên cột, dữ liệu tương ứng và nạp câu truy vấn:

‘**INSERT INTO TÊN\_BẢNG (TÊN\_CỘT\_1, TÊN\_CỘT\_2,...)**  
**VALUES (DỮ\_LIỆU\_CỘT\_1, DỮ\_LIỆU\_CỘT\_2,...)**’.

```
function insert($tableName, $data)
{
    global $conn;

    $table = validate($tableName);

    $keys = array_keys($data);
    $values = array_values($data);

    $keys = implode('`', $keys);
    $values = "``".implode('`', $values)."``";

    $query = "INSERT INTO $table ($keys) VALUES ($values)";

    $result =
    [ 'status' => sqlsrv_query($conn, $query),
      'query' => $query,
    ];
    return $result;
}
```

- Hàm cập nhật dữ liệu với dữ liệu truyền vào là tên bảng, điều kiện dữ liệu (key, value) và mảng kết hợp : “tên cột” => “dữ liệu”. Hàm thực hiện lấy ra tên cột, dữ liệu tương ứng và nạp câu truy vấn :

#### ‘UPDATE TÊN\_BẢNG

```
SET TÊN_CỘT_1 = DỮ_LIỆU_CỘT_1,  
    TÊN_CỘT_2 = DỮ_LIỆU_CỘT_2,...
```

```
WHERE $key = $value’.
```

```
function updatebyKeyValue($tableName, $keys, $values, $data)  
{  
    global $conn;  
    //print_r($data);  
    $table = validate($tableName);  
    $keys = validate($keys);  
    $values = validate($values);  
  
    $dataString = "";  
  
    //" key=value,"  
    foreach($data as $key => $value)  
    {  
        $dataString .= "$key = '$value', ";  
    }  
  
    $setClause = rtrim($dataString, ', ');  
  
    $query = "UPDATE $table SET $setClause WHERE $keys ='$values'";  
  
    //echo $query;  
    $result = [  
        'status' => sqlsrv_query($conn, $query),  
        'query'  => $query,  
    ];  
  
    return $result;  
}
```

- Hàm thực hiện lấy dữ liệu từ bảng có tích hợp kỹ thuật phân trang với dữ liệu truyền vào là tên bảng cần lấy dữ liệu, số dòng dữ liệu trong một trang, số thứ tự trang hiện tại và thứ tự sắp xếp dữ liệu. Đầu tiên hàm thực hiện tính toán dòng đầu tiên của trang dữ liệu dựa vào số thứ tự của trang dữ liệu và nạp câu truy vấn:

```
“SELECT * FROM $table ORDER BY $pageOrder  
OFFSET $startRow ROWS  
FETCH NEXT $pageSize ROWS ONLY”
```

```
function getAllWithPagination($tableName, $pageSize, $pageNumber, $pageOrder)  
{  
    global $conn;  
  
    $table = validate($tableName);  
    $startRow = ($pageNumber - 1) * $pageSize;  
  
    $query = "SELECT COUNT(*) as total FROM $table"; // Đếm tổng số dòng  
    $result = sqlsrv_query($conn, $query);  
    $totalRows = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)[ 'total' ];  
  
    $query = "SELECT * FROM $table ORDER BY $pageOrder OFFSET $startRow ROWS FETCH NEXT $pageSize ROWS ONLY";  
    $result = sqlsrv_query($conn, $query);  
  
    if ($result) {  
        $data = array();  
  
        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {  
            $data[] = $row;  
        }  
  
        $response = [  
            'status' => 'Data Found',  
            'data' => $data,  
            'total' => $totalRows,  
        ];  
    } else {  
        $response = [  
            'status' => 'Something went wrong! Please try again.',  
        ];  
    }  
  
    return $response;  
}
```

- Hàm lấy dữ liệu dựa trên điều kiện ‘key’ = ‘value’. Dữ liệu truyền vào là tên bảng và điều kiện. Câu truy vấn tương ứng:

“**SELECT \* FROM \$table WHERE \$key = \$value**”.

```
function getAllByKeyValue($tableName, $key, $value)
{
    global $conn;

    $table = validate($tableName);

    $query = "SELECT * FROM $table WHERE $key = '$value'";
    $result = sqlsrv_query($conn, $query);

    if ($result) {
        $data = array();

        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
            $data[] = $row;
        }

        if (!empty($data)) {
            $response = [
                'status' => 'Data Found',
                'data' => $data,
            ];
        } else {
            $response = [
                'status' => 'No Data Found',
            ];
        }

        return $response;
    } else {
        $response = [
            'status' => 'Something went wrong! Please try again.',
        ];
        return $response;
    }
}
```

- Hàm xem cuộc hẹn gần nhất với dữ liệu truyền vào là số dòng dữ liệu cần lấy. Ví dụ, ta cần xem 10 cuộc hẹn gần nhất thì gọi hàm `getTopAppt(10)`, hàm thực hiện lấy ra 10 dòng dữ liệu đầu tiên từ bảng APPOINTMENT với thứ tự ngày và giờ gần nhất, câu truy vấn tương ứng:

“`SELECT TOP 10 * FROM APPOINTMENT ORDER BY Date_Appt DESC, Time_Appt DESC`”.

```
function getTopAppt($count)
{
    global $conn;

    $count = validate($count);

    $query = "SELECT TOP $count *
              FROM APPOINTMENT
              ORDER BY Date_Appt DESC, Time_Appt DESC";
    $result = sqlsrv_query($conn, $query);

    if ($result) {
        $data = array();

        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
            $data[] = $row;
        }

        if (!empty($data)) {
            $response = [
                'status' => 'Data Found',
                'data' => $data,
            ];
        } else {
            $response = [
                'status' => 'No Data Found',
            ];
        }

        return $response;
    } else {
        $response = [
            'status' => 'Something went wrong! Please try again.',
        ];
        return $response;
    }
}
```

- Hàm xem cuộc hẹn gần nhất với dữ liệu truyền vào là số dòng dữ liệu cần lấy. Ví dụ, ta cần xem 100 hóa đơn gần nhất thì gọi hàm `getTopInvoice(100)`, hàm thực hiện lấy ra 100 dòng dữ liệu đầu tiên từ bảng INVOICE với thứ tự ngày và giờ gần nhất, câu truy vấn tương ứng:

“`SELECT TOP 100 * FROM INVOICE ORDER BY InvoiceTime DESC`”.

```
function getTopInvoice($count)
{
    global $conn;

    $count = validate($count);

    $query = "SELECT TOP $count *
              FROM INVOICE
              ORDER BY InvoiceTime DESC";
    $result = sqlsrv_query($conn, $query);

    if ($result) {
        $data = array();

        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
            $data[] = $row;
        }

        if (!empty($data)) {
            $response = [
                'status' => 'Data Found',
                'data' => $data,
            ];
        } else {
            $response = [
                'status' => 'No Data Found',
            ];
        }

        return $response;
    } else {
        $response = [
            'status' => 'Something went wrong! Please try again.',
        ];
        return $response;
    }
}
```

- Hàm xóa dòng dữ liệu dựa trên điều kiện ‘key’ = ‘value’. Dữ liệu truyền vào là tên bảng và điều kiện. Câu truy vấn tương ứng:

“**DELETE FROM \$table WHERE \$key = \$value** ”.

```
function delete($tableName, $key, $value)
{
    global $conn;

    $table = validate($tableName);
    $key = validate($key);
    $value = validate($value);

    $query = "DELETE FROM $table WHERE $key = '$value'";

    $result = [
        'status' => sqlsrv_query($conn, $query),
        'query' => $query
    ];

    return $result;
}
```

- Hàm lấy tên các cột trong bảng đã cho. Câu truy vấn tương ứng :

“**SELECT COLUMN\_NAME FROM INFORMATION\_SCHEMA.COLUMNS WHERE TABLE\_NAME = \$tableName** ”.

```
// Hàm lấy tên các cột trong bảng
function getTableColumns($tableName)
{
    global $conn;

    $tableName = validate($tableName);

    // Truy vấn để lấy tên các cột
    $query = "SELECT COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME = ?";

    // Thực thi truy vấn
    $params = array($tableName);
    $result = sqlsrv_query($conn, $query, $params);

    $columns = array();

    while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
        $columns[] = $row['COLUMN_NAME'];
    }

    return $columns;
}
```

- Hàm tìm kiếm dựa trên từ khóa với dữ liệu truyền vào là từ khóa và tên bảng cần tìm kiếm. Ví dụ, ta cần tìm kiếm với từ khóa ‘Levi’ trong bảng USER\_DENTAL thì gọi hàm searchByKeyword(‘USER\_DENTAL’, ‘Levi’), câu truy vấn tương ứng là :

```
'SELECT * FROM USER_DENTAL
WHERE (ID_User LIKE '%Levi%' OR ID LIKE '%Levi%' OR
Username LIKE '%Levi%' OR Fullname LIKE '%Levi%' OR
Gender LIKE '%Levi%' OR PhoneNumber LIKE '%Levi%' OR
CurrAddress LIKE '%Levi%' OR UserType LIKE '%Levi%)'
```

```
function searchByKeyword($tableName, $value)
{
    global $conn;

    $table = validate($tableName);
    $value = validate($value);

    $query = "SELECT * FROM $table WHERE ";
    $columns = getTableColumns($tableName);

    $conditions = [];
    foreach ($columns as $column) {
        $conditions[] = "$column LIKE '%$value%'";
    }

    $query = "SELECT * FROM $tableName WHERE ";
    $query = str_pad($query, strlen($query) + 1, "(");
    $query .= implode(" OR ", $conditions);
    $query = str_pad($query, strlen($query) + 1, ")");

    // Thực hiện truy vấn chính
    $result = sqlsrv_query($conn, $query);

    if ($result) { ...
    } else {
        $response = [ ...
        ];
        return $response;
    }
}
```

- Hàm thực hiện lấy tên người dùng và ID dựa trên loại người dùng với dữ liệu truyền vào là tên bảng và loại người dùng (Admin, Nhân viên, Nha sĩ). Ví dụ, ta cần lấy ra danh sách các nha sĩ có trong phòng khám thì gọi hàm `getIdbyUserType('USER_DENTAL', 'Dentist')`, câu truy vấn tương ứng :

```
SELECT ID_User, Fullname FROM USER_DENTAL WHERE UserType = 'Dentist'
```

```
function getIdbyUserType($tableName, $userType)
{
    global $conn;

    $table = validate($tableName);
    $userType = validate($userType);

    $query = "SELECT ID_User, Fullname FROM $table WHERE UserType = '$userType'";
    $result = sqlsrv_query($conn, $query);

    if ($result) {
        $data = [];

        while ($row = sqlsrv_fetch_array($result, SQLSRV_FETCH_ASSOC)) {
            $data[] = $row;
        }

        if (!empty($data)) {
            $response = [
                'status' => 'Data Found',
                'data' => $data,
            ];
        } else {
            $response = [
                'status' => 'No Data Found',
            ];
        }

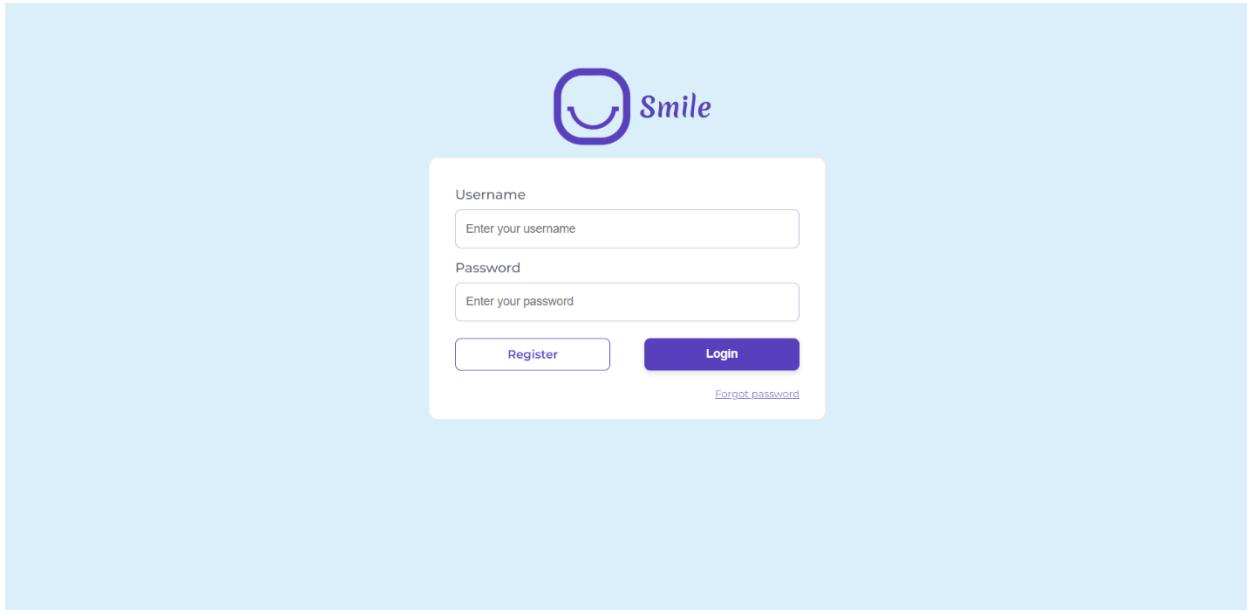
        return $response;
    } else {
        $response = [
            'status' => 'Something went wrong! Please try again.',
        ];
        return $response;
    }
}
```

### VIII. DEMO ỨNG DỤNG

#### 1. Chức năng chính trong ứng dụng:

##### 1.1. Đăng nhập:

- **Admin, nhân viên, nha sĩ:** Đăng nhập vào website bằng cách nhập Username và Password.



- Sau khi đăng nhập thành công, người dùng sẽ được chuyển hướng đến dashboard, và xem được 10 cuộc hẹn và 10 hóa đơn gần nhất.

The screenshot displays the System Admin Dashboard interface. On the left, there is a sidebar with a logo and navigation links: Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main area is titled "SYSTEM ADMIN DASHBOARD". It contains two tables: "Recent Appointments" and "Recent Invoice".

**Recent Appointments**

APPOINTMENT ID	CUSTOMER	DENTIST	ROOM	DATE	TIME	TYPE
AP00000053	6	12	R14	30-12-2023	08:00	NEW
AP00000052	4	14	R13	20-12-2023	13:40	NEW
AP00000002	16	4769	R13	03-12-2023	22:37	REASSESS
AP00000012	15	2706	R4	03-12-2023	13:18	NEW
AP00000010	18	6230	R9	28-11-2023	16:07	REASSESS
AP00000026	7	7636	R3	26-11-2023	13:01	REASSESS
AP00000023	3	2123	R1	14-11-2023	18:51	REASSESS
AP00000014	11	2456	R13	09-11-2023	22:49	REASSESS
AP00000009	16	7269	R8	01-11-2023	14:39	REASSESS
AP00000040	7	8521	R3	30-10-2023	16:28	NEW

**Recent Invoice**

INVOICE ID	TREATMENT PLAN	PAYMENT ID	TOTAL (\$)	TIME
U000000051	51	Cash	0	20:30:23 2023-12-24
U000000033	33	Momo	0	19:40:12 2023-12-20
U000000034	34	Momo	14129	19:40:12 2023-12-20
U000000035	35	Momo	14099	19:40:12 2023-12-20
U000000036	36	Momo	1864	19:40:12 2023-12-20
U000000037	37	Momo	0	19:40:12 2023-12-20
U000000038	38	Cash	0	19:40:12 2023-12-20
U000000039	39	Momo	521	19:40:12 2023-12-20
U000000040	40	Momo	8874	19:40:12 2023-12-20
U000000041	41	Cash	1161	19:40:12 2023-12-20

### 1.2. Thông tin cá nhân:

- **Admin, nhân viên và nha sĩ:** Có thể xem và chỉnh sửa thông tin cá nhân của mình.

The screenshot shows the 'SYSTEM ADMIN DASHBOARD' for a user named Aleida Goforth. The interface includes a sidebar with navigation links for Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, and Invoices, along with a Log Out button. The main content area displays a 'Hello Aleida Goforth' greeting and a message stating, 'This is your profile page. You can customize your profile as you want and also change password too'. Below this, there are two sections for 'My Account' and 'USER INFORMATION'. Each section contains fields for Fullname (Aleida Goforth), Phone Number (51167), Gender (Male), and Address (2709 Monument Parkway). There is also a 'CHANGE PASSWORD' section with a Password field and a 'Change Profile' button. On the right side of the dashboard, there are two cards for 'System Admin Admin1' and 'System Admin Admin1', each featuring a blue circular profile icon and a smiley face icon below it. The footer of the dashboard indicates the date as December, 2023 - AdvDB - HCMUS and features a 'Smile' logo.

### 1.3. Danh sách bệnh nhân:

- Admin, nhân viên và nha sĩ: Xem được danh sách bệnh nhân trong tab Patients.

PATIENT ID	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	ACTION
1	Dniren Sweatman	Female	5861161688	413 Straubel Avenue	<button>View Detail</button>
2	Torrin Yushin	Female	9428484633	80 Eggendart Junction	<button>View Detail</button>
3	Hollie Duffill	Male	6451007830	9226 Laurel Court	<button>View Detail</button>
4	Shep Murrock	Female	9171475127	427 Dottie Drive	<button>View Detail</button>
5	Elvin Gyurkovics	Male	3481759710	68 Lakewood Gardens Lane	<button>View Detail</button>
6	Adam Say	Male	2926448299	305 Walton Plaza	<button>View Detail</button>
7	Cosmo Tombleson	Female	8994031782	83903 Forest Road	<button>View Detail</button>
8	Francene Moring	Female	8319650584	1 Nobel Pass	<button>View Detail</button>

- Admin, nhân viên và nha sĩ: Có thể nhập thông tin cần tìm kiếm và nhấn “Search”, hệ thống sẽ tìm kiếm trên tất cả các trường dữ liệu.

PATIENT ID	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	ACTION
2	Torrin Yushin	Female	9428484633	80 Eggendart Junction	<button>View Detail</button>
9034	Torrin Ranscombe	Female	0662241653	1203 Waubesa Park	<button>View Detail</button>
14868	Torrin Kittless	Female	0591738514	9244 Sundown Point	<button>View Detail</button>
24860	Torrin Ivakhnov	Female	0660792385	64 Eliot Way	<button>View Detail</button>
54779	Torrin Burrell	Female	0184343591	42 Eagle Crest Street	<button>View Detail</button>
76338	Torrin Kowalik	Female	0588837191	7 Wayridge Hill	<button>View Detail</button>
93220	Torrin Mutter	Male	0381413661	4 Reindahl Way	<button>View Detail</button>
99463	Torrin Dumberell	Male	0638810363	67072 Blue Bill Park Park	<button>View Detail</button>

- Admin, nhân viên và nha sĩ:** Sau khi người dùng nhấn “Add new patient”, có thể thêm bệnh nhân mới bằng cách điền thông tin vào form, và nhấn “Add Patient” để thêm mới bệnh nhân.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Patient Name	Gender	Patient Phone Number
Enter your name	Male	Enter your phone number

Patient Address	Date Of Birth
Enter your address	dd/mm/yyyy

Add Patient

December, 2023 - AdvDB - HCMUS

Smile

- Admin, nhân viên và nha sĩ:** Sau khi người dùng nhấn “View detail”, có thể xem thông tin chi tiết của bệnh nhân, bao gồm:
  - Cập nhật thông tin bệnh nhân: Chính sửa thông tin và nhấn “Update Patient” để cập nhật.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Patient Name	Gender	Phone Number
Dniren Sweatman	Female	5861161688

Patient Address	Date Of Birth
413 Straubel Avenue	19/03/1997

Update Patient

Tooth Problem

PATIENT ID	NOTE DATE	DESCRIPTION	ACTION
1	2005-07-11 12:33	Receding Gums	<span style="color: red;">Delete</span>

Add

- Xem vấn đề răng của bệnh nhân đó: Người dùng có thể nhập mô tả và nhấn “Add” để thêm vấn đề răng cho bệnh nhân, hoặc nhấn “Delete” để xóa vấn đề tương ứng.

- Xem những loại thuốc chống chỉ định cho bệnh nhân: Người dùng có thể nhập ID thuốc mà chống chỉ định cho bệnh nhân và nhấn “Add” để thêm, tương tự, nhấn “Delete” để xóa loại thuốc tương ứng

Tooth Problem			
PATIENT ID	NOTE DATE	DESCRIPTION	ACTION
1	2005-07-11 12:33	Receding Gums	<button>Delete</button>
1	2023-12-24 16:45	Root Infection	<button>Delete</button>

Contraindication		
PATIENT ID	MEDICINE ID	ACTION
1	12	<button>Delete</button>
1	24	<button>Delete</button>
1	153	<button>Delete</button>

- Xem kế hoạch điều trị: Người dùng có thể xem danh sách kế hoạch điều trị cho bệnh nhân.

Treatment Plan						
PATIENT ID	SELECT TREATMENT ID	DENTIST ID	DATETIME	RETURN DAYS	STATUS	ACTIONS
1	14836	1	2014-02-02 00:00:00	90	COMPLETED	<button>View detail</button>
1	33307	1	2020-03-01 00:00:00	46	PLANNING	<button>View detail</button>
1	70384	1	2006-04-02 00:00:00	28	COMPLETED	<button>View detail</button>
1	101008	8	2023-12-20 17:38:00	12	PLANNING	<button>View detail</button>
1	101009	1	2023-12-20 17:38:00	12	COMPLETED	<button>View detail</button>

December, 2023 - AdvDB - HCMUS

Smile

- + Nhấn “Add new plan” để thêm kế hoạch điều trị mới, điền vào các thông tin tương ứng, nhấn “Add Treatment Plan” để thêm.

The screenshot shows the 'SYSTEM ADMIN DASHBOARD' interface. On the left is a sidebar with a logo and navigation links: Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main area has a header 'SYSTEM ADMIN DASHBOARD' and a user profile 'Aleida Goforth'. A message 'Please Fill All Fields' is displayed above a form. The form includes fields for 'Patient Id' (1), 'Dentist Id' (Patrick Perales (ID\_Dentist = 1)), 'Date Time' (dd/mm/yyyy - - : -- --), 'Return Days' (empty), and a 'Add Treatment Plan' button. At the bottom, it shows 'December, 2023 - AdvDB - HCMUS' and 'Smile'.

- + Nhấn “View detail” để xem chi tiết kế hoạch điều trị, người dùng có thể cập nhật thông tin điều trị; xem, thêm, xóa loại răng và dịch vụ cho kế hoạch chữa trị đó.

The screenshot shows the 'SYSTEM ADMIN DASHBOARD' interface. The sidebar and header are identical to the previous screenshot. The main area displays a 'Please Fill All Fields' message and a form for updating a treatment plan. The form includes fields for 'Select Treatment Id' (14836), 'Patient Name' (Driren Sweatman), 'Dentist Name' (Patrick Perales (ID\_Dentist = 1)), 'Date Time' (02/02/2014 12:00 SA), 'Return Days' (90), and 'Select Treatment Status' (Completed). Below this is a 'Choose Tooth' section with a table:

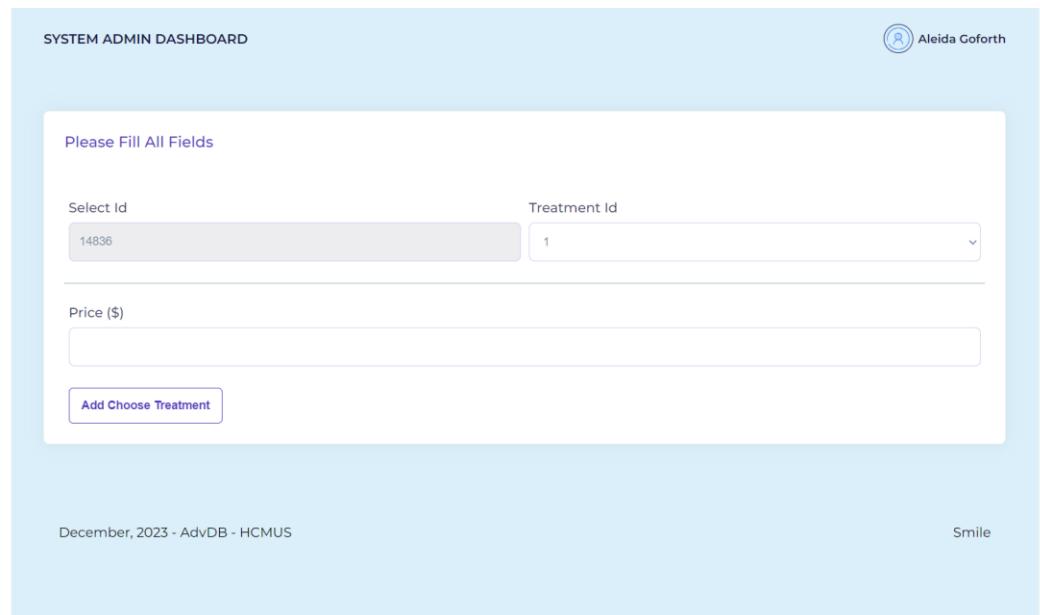
SELECT ID	TOOTH ID	SURFACE	PRICE (\$)	ACTION
14836	1	Lingual	12	<span style="color: red;">Delete</span>

The screenshot shows a dashboard for the Smile dental management system. On the left, there's a sidebar with navigation links: Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main area has two tables. The first table, titled 'Choose Tooth', has columns: SELECT ID, TOOTH ID, SURFACE, PRICE (\$), and ACTION. It contains one row with values: 14836, 1, Lingual, 12, and a red 'Delete' button. The second table, titled 'Choose Treatment', has columns: SELECT ID, TREATMENT ID, PRICE (\$), and ACTION. It contains one row with values: 14836, 1, 20, and a red 'Delete' button. At the bottom left is the text 'December, 2023 - AdvDB - HCMUS'. At the bottom right is the word 'Smile'.

- + Thêm răng cần chữa trị.

The screenshot shows a 'SYSTEM ADMIN DASHBOARD' for the Smile dental management system. On the left, there's a sidebar with navigation links: Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main area has a form titled 'Please Fill All Fields' with fields for Select Id, Tooth Id, Surface Id, and Price (\$). The 'Select Id' field contains '14836', the 'Tooth Id' field contains '1', the 'Surface Id' field contains 'Lingual', and the 'Price (\$)' field is empty. Below the form is a button labeled 'Add Choose Tooth'. At the bottom left is the text 'December, 2023 - AdvDB - HCMUS'. At the bottom right is the word 'Smile'.

- + Thêm dịch vụ điều trị.



#### 1.4. Danh sách nhân viên:

- Admin:** Xem được danh sách nhân viên sau khi nhấp vào tab Staffs, có thể xóa nhân viên tương ứng khi nhấp “Delete”.

The screenshot shows a 'SYSTEM ADMIN DASHBOARD' interface with a sidebar and a main staff list table. The table has columns: ID, USERNAME, FULL NAME, GENDER, PHONE NUMBER, ADDRESS, STATUS, and ACTION. The data in the table is as follows:

ID	USERNAME	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	STATUS	ACTION
U000000021	Nhanvien1	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000022	Nhanvien2	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000023	Nhanvien3	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000024	Nhanvien4	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000025	Nhanvien5	Patrick Perales	Female	76272	13 Edgewood Dr	DELETED	
U000000026	Nhanvien6	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000027	Nhanvien7	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	
U000000028	Nhanvien8	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	

- Admin:** Sau khi nhấn “Add new staff”, có thể thêm nhân viên mới bằng cách điền thông tin vào form, và nhấn “Add Staff” để thêm mới nhân viên.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Staff Name  Gender

Staff Address  Staff Phone Number

Username  Password

**Add Staff**

- Admin:** Cập nhật thông tin nhân viên: Chính sửa thông tin và nhấn “Update Staff” để cập nhật.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Staff Name  Gender

Staff Address  Staff Phone Number

Username  Password

**Update Staff**

- Nhân viên và nha sĩ:** Xem được danh sách nhân viên sau khi nhấp vào tab Staffs.

The screenshot shows the 'SYSTEM STAFF DASHBOARD' interface. On the left is a sidebar with a 'Smile' logo and links for Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main area is titled 'Staff Records' and displays a table with 10 rows of staff information. Each row includes columns for ID, Username, Full Name, Gender, Phone Number, Address, and Status. The status column for the 5th row ('Nhanvien5') is 'DELETED' with a red background. The other rows have a green 'ACTIVE' status.

ID	USERNAME	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	STATUS
U0000000021	Nhanvien1	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000022	Nhanvien2	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000023	Nhanvien3	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000024	Nhanvien4	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000025	Nhanvien5	Patrick Perales	Female	76272	13 Edgewood Dr	DELETED
U0000000026	Nhanvien6	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000027	Nhanvien7	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000028	Nhanvien8	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000029	Nhanvien9	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U0000000030	Nhanvien10	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE

### 1.5. Danh sách nha sĩ:

- Admin:** Tương tự với trang nhân viên, Admin có thể thêm, xóa, sửa thông tin nha sĩ.

The screenshot shows the 'SYSTEM ADMIN DASHBOARD' interface. The sidebar is identical to the staff dashboard. The main area is titled 'DENTIST RECORDS' and displays a table with 8 rows of dentist information. Each row includes columns for Dentist ID, Username, Full Name, Gender, Phone Number, Address, Status, and Action. The status column for all rows is 'ACTIVE'. Each row has a red 'Delete' button in the action column.

DENTIST ID	USERNAME	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	STATUS	ACTION
U0000000001	Nhasi1	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000002	Nhasi2	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000003	Nhasi3	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000004	Nhasi4	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000005	Nhasi5	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000006	Nhasi6	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000007	Nhasi7	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>
U0000000008	Nhasi8	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE	<span style="color:red;">Delete</span>



- [!\[\]\(2e57ef66acf4c05ebf0183fc980489ea\_img.jpg\) Dashboard](#)
- [!\[\]\(58f356ef582c82e69228e0267d44c27c\_img.jpg\) Patients](#)
- [!\[\]\(946e7d08dc14d4455ac57eab106ac0ad\_img.jpg\) Staffs](#)
- [!\[\]\(ad19c0ec8c364c24b1ce4264cda1fed6\_img.jpg\) Dentists](#)
- [!\[\]\(de25ef926c4cd97e683fbc6cdc28d27a\_img.jpg\) Medicines](#)
- [!\[\]\(ae3368e45f38515f19ab6fce60fd459b\_img.jpg\) Appointments](#)
- [!\[\]\(a50658340fe76a67b1c0fd767e3dbf8b\_img.jpg\) Invoices](#)

---

[!\[\]\(3e14947cf97289533868d4cb08fb9d34\_img.jpg\) Log Out](#)

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Dentist Name	Gender
<input type="text" value="Enter your fullname"/>	<input type="text" value="Male"/>
<hr/>	
Dentist Address	Dentist Phone Number
<input type="text" value="Enter your address"/>	<input type="text" value="Enter your phone number"/>
<hr/>	
Username	Password
<input type="text" value="Enter your username"/>	<input type="text" value="Enter your password"/>
<input type="button" value="Add Dentist"/>	

Aleida Coforth

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Dentist Name	Gender
<input type="text" value="Patrick Perales"/>	<input type="text" value="Female"/>
<hr/>	
Dentist Address	Phone Number
<input type="text" value="13 Edgewood Dr"/>	<input type="text" value="76272"/>
<hr/>	
Username	Password
<input type="text" value="Nhasl1"/>	<input type="text"/>
<input type="button" value="Update Dentist"/>	

Aleida Coforth

- **Nhân viên và nha sĩ:** Xem danh sách thông tin nha sĩ.

The screenshot shows a dashboard titled "SYSTEM STAFF DASHBOARD". On the left is a sidebar with a "Smile" logo and links for Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, and Invoices, along with a Log Out button. The main area is titled "Dentist Records" and contains a table with the following data:

DENTIST ID	USERNAME	FULL NAME	GENDER	PHONE NUMBER	ADDRESS	STATUS
U000000001	Nhasil	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000002	Nhasi2	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000003	Nhasi3	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000004	Nhasi4	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000005	Nhasi5	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000006	Nhasi6	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000007	Nhasi7	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000008	Nhasi8	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000009	Nhasi9	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE
U000000010	Nhasi10	Patrick Perales	Female	76272	13 Edgewood Dr	ACTIVE

## 1.6. Danh sách thuốc:

- **Admin:** Có thể xem, thêm và chỉnh sửa thông tin thuốc.

The screenshot shows a dashboard titled "SYSTEM ADMIN DASHBOARD". On the left is a sidebar with a "Smile" logo and links for Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, and Invoices, along with a Log Out button. The main area is titled "Medicines" and contains a table with the following data:

MEDICINE ID	MEDICINE NAME	PRICE (\$)	ACTION
ME00000001	Ranitidine Hydrochloride	840	<button>Add new medicine</button> <button>Update</button>
ME00000002	Terbinafine Hydrochloride	490	<button>Update</button>
ME00000003	dapsone	315	<button>Update</button>
ME00000004	Glipizide	664	<button>Update</button>
ME00000005	Stannous Fluoride	118	<button>Update</button>
ME00000006	Divalproex Sodium Delayed Release	661	<button>Update</button>
ME00000007	triazolam	263	<button>Update</button>
ME00000008	Titanium Dioxide and Octinoxate	349	<button>Update</button>

- **Admin:** Thêm thuốc mới bằng cách nhập tên và giá thuốc, nhấn “Add Medicine” để thêm.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Medicine Name: Enter medicine name

Medicine Price (\$): Enter medicine price

Add Medicine

December, 2023 - AdvDB - HCMUS

Aleida Goforth

Smile

- **Admin:** Chính sửa thông tin gồm tên và giá thuốc, nhấn “Update Medicine” để cập nhật.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Medicine Name: Ranitidine Hydrochloride

Medicine Price (\$): 840

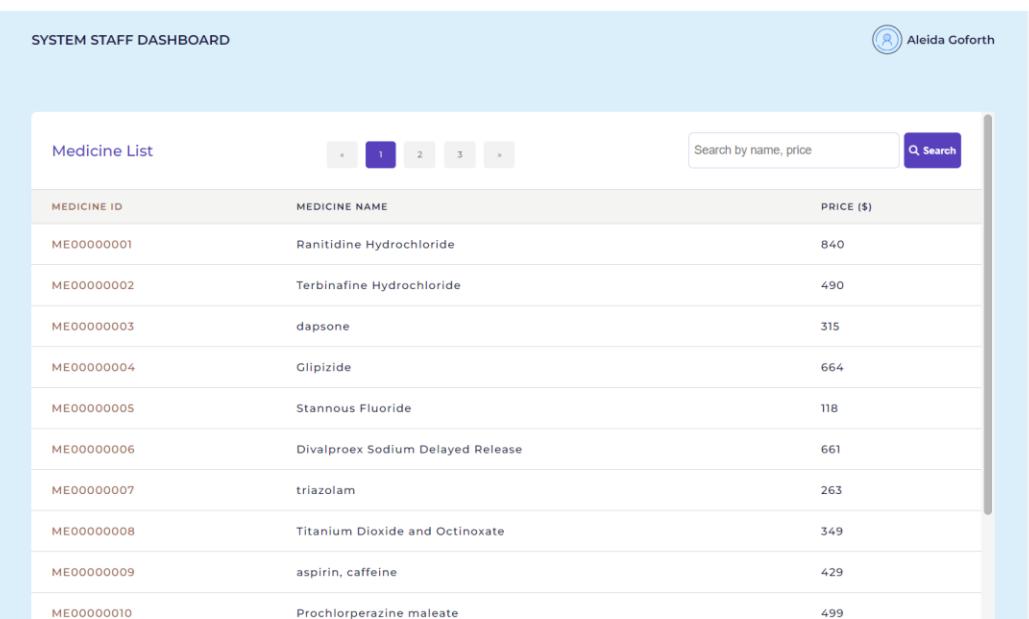
Update Medicine

December, 2023 - AdvDB - HCMUS

Aleida Goforth

Smile

- **Nhân viên và nha sĩ:** Xem danh sách thuốc.

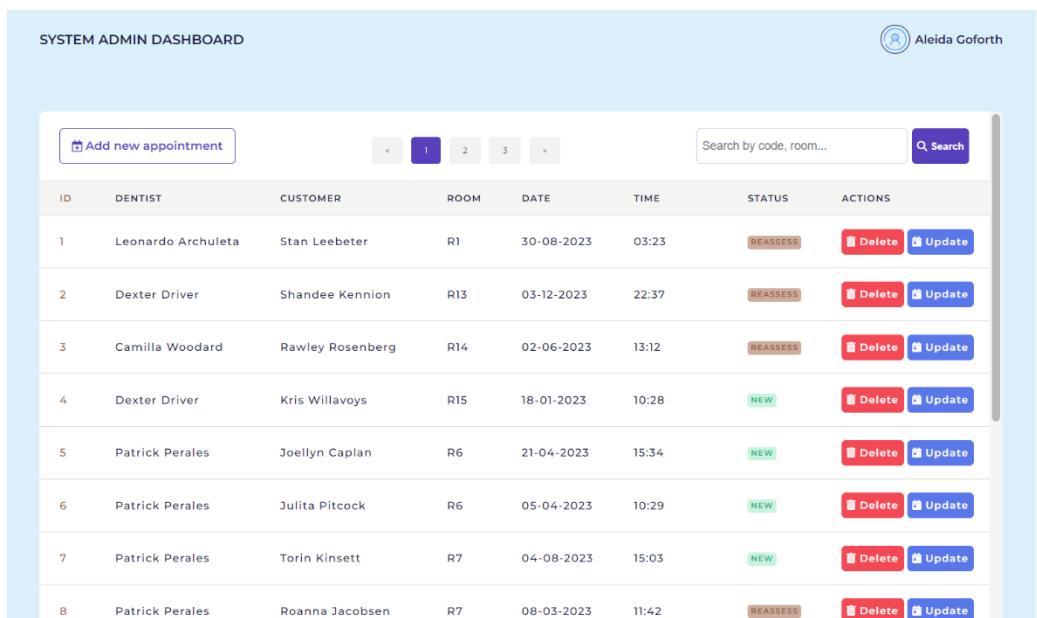


The screenshot shows the 'SYSTEM STAFF DASHBOARD' with a 'Medicine List' table. The table has columns for MEDICINE ID, MEDICINE NAME, and PRICE (\$). The data is as follows:

MEDICINE ID	MEDICINE NAME	PRICE (\$)
ME00000001	Ranitidine Hydrochloride	840
ME00000002	Terbinafine Hydrochloride	490
ME00000003	dapsone	315
ME00000004	Glipizide	664
ME00000005	Stannous Fluoride	118
ME00000006	Divalproex Sodium Delayed Release	661
ME00000007	triazolam	263
ME00000008	Titanium Dioxide and Octinoxate	349
ME00000009	aspirin, caffeine	429
ME00000010	Prochlorperazine maleate	499

### 1.7. Danh sách cuộc hẹn:

- **Admin, nhân viên:** Có thể xem, thêm và sửa thông tin cuộc hẹn.



The screenshot shows the 'SYSTEM ADMIN DASHBOARD' with an 'Appointments' table. The table has columns for ID, DENTIST, CUSTOMER, ROOM, DATE, TIME, STATUS, and ACTIONS. The data is as follows:

ID	DENTIST	CUSTOMER	ROOM	DATE	TIME	STATUS	ACTIONS
1	Leonardo Archuleta	Stan Leebeter	R1	30-08-2023	03:23	REASSESS	<button>Delete</button> <button>Update</button>
2	Dexter Driver	Shandee Kennion	R13	03-12-2023	22:37	REASSESS	<button>Delete</button> <button>Update</button>
3	Camilla Woodard	Rawley Rosenberg	R14	02-06-2023	13:12	REASSESS	<button>Delete</button> <button>Update</button>
4	Dexter Driver	Kris Willavoys	R15	18-01-2023	10:28	NEW	<button>Delete</button> <button>Update</button>
5	Patrick Perales	Joellyn Caplan	R6	21-04-2023	15:34	NEW	<button>Delete</button> <button>Update</button>
6	Patrick Perales	Julita Pitcock	R6	05-04-2023	10:29	NEW	<button>Delete</button> <button>Update</button>
7	Patrick Perales	Torin Kinsett	R7	04-08-2023	15:03	NEW	<button>Delete</button> <button>Update</button>
8	Patrick Perales	Roanna Jacobsen	R7	08-03-2023	11:42	REASSESS	<button>Delete</button> <button>Update</button>

- Admin, nhân viên:** Thêm cuộc hẹn mới bằng cách nhập vào form tương ứng, nhấn “Add Appointment” để thêm.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Dentist Name	Patient Id	Room
Patrick Perales (ID_Dentist = 1)		R1 (Floor = 1)

Appointment Date	Appointment Time	Appointment Status
dd/mm/yyyy	-- : -- : --	New

Add Appointment

December, 2023 - AdvDB - HCMUS

Smile

- Admin:** Chính sửa thông tin cuộc hẹn, nhấn “Update Appointment” để cập nhật.

SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Dentist Name	Patient Name	Room
Leonardo Archuleta (ID_Dentist = 14)	Stan Leebeeter	R1 (Floor = 1)

Appointment Date	Appointment Time	Appointment Status
30/08/2023	03:23 SA	Reassess

Update Appointment

December, 2023 - AdvDB - HCMUS

Smile

- **Nha sĩ:** Có thể xem thông tin cuộc hẹn.

The screenshot shows the 'SYSTEM DENTIST DASHBOARD' with a header featuring a user profile icon and the name 'Aleida Goforth'. On the left is a sidebar with a logo, navigation links (Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices), and a Log Out button. The main area displays an 'Appointment List' table with columns: ID, DENTIST, CUSTOMER, ROOM, DATE, TIME, and STATUS. The table contains 10 rows of data, each with a 'REASSESS' button. A search bar at the top right allows searching by code, room, and date.

ID	DENTIST	CUSTOMER	ROOM	DATE	TIME	STATUS
1	Leonardo Archuleta	Stan Leebeter	R1	30-08-2023	03:23	REASSESS
2	Dexter Driver	Shandee Kennion	R13	03-12-2023	22:37	REASSESS
3	Camilla Woodard	Rawley Rosenberg	R14	02-06-2023	13:12	REASSESS
4	Dexter Driver	Kris Willavos	R15	18-01-2023	10:28	NEW
5	Patrick Perales	Joellyn Caplan	R6	21-04-2023	15:34	NEW
6	Patrick Perales	Julita Pitcock	R6	05-04-2023	10:29	NEW
7	Patrick Perales	Torin Kinsett	R7	04-08-2023	15:03	NEW
8	Patrick Perales	Roanna Jacobsen	R7	08-03-2023	11:42	REASSESS
9	Dexter Driver	Ingunna Bourbon	R8	01-11-2023	14:39	REASSESS
10	Gregorio Bostick	Consolata McGeffen	R9	28-11-2023	16:07	REASSESS

### 1.8. Danh sách hóa đơn:

- **Admin, nhân viên:** Có thể xem, thêm và sửa thông tin hóa đơn.

The screenshot shows the 'SYSTEM ADMIN DASHBOARD' with a header featuring a user profile icon and the name 'Aleida Goforth'. On the left is a sidebar with a logo, navigation links (Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices), and a Log Out button. The main area displays an 'Invoice List' table with columns: INVOICE ID, SELECT TREATMENT, PAYMENT, TOTAL (\$), TIME, and ACTIONS. The table contains 8 rows of data, each with a 'View Detail' button. A search bar at the top right allows searching by code, date, time, and treatment.

INVOICE ID	SELECT TREATMENT	PAYMENT	TOTAL (\$)	TIME	ACTIONS
U000000001	1	Momo	20635	19:40:12 2023-12-20	<button>View Detail</button>
U000000002	2	Momo	39507	19:40:12 2023-12-20	<button>View Detail</button>
U000000003	3	Momo	15793	19:40:12 2023-12-20	<button>View Detail</button>
U000000004	4	Cash	20361	19:40:12 2023-12-20	<button>View Detail</button>
U000000005	5	Cash	12119	19:40:12 2023-12-20	<button>View Detail</button>
U000000006	6	Momo	3260	19:40:12 2023-12-20	<button>View Detail</button>
U000000007	7	Cash	3702	19:40:12 2023-12-20	<button>View Detail</button>
U000000008	8	Momo	27034	19:40:12 2023-12-20	<button>View Detail</button>

- Admin, nhân viên:** Nhấn “Add new invoice” để tạo hóa đơn, nhập thông tin tương ứng và nhấn “Add Invoice” để thêm hóa đơn.

POLYCLINIC SYSTEM ADMIN DASHBOARD

Please Fill All Fields

Treatment Plan Id	Payment	
	Cash	
Tooth Price	Medicine Price	Invoice Total
Amount Paid	Change	
0		

Add Invoice

- Nha sĩ:** Xem danh sách hóa đơn.

POLYCLINIC SYSTEM DENTIST DASHBOARD

Invoice List

INVOICE ID	SELECT TREATMENT	PATIENT NAME	PAYMENT	TOTAL (\$)	TIME	ACTIONS
U000000001	1	Juieta Bracci	Momo	20635	19:40:12 2023-12-20	<button>View Detail</button>
U000000002	2	Marilee Dilliway	Momo	39507	19:40:12 2023-12-20	<button>View Detail</button>
U000000003	3	Lindsay Janton	Momo	15793	19:40:12 2023-12-20	<button>View Detail</button>
U000000004	4	Adolphe Spurrior	Cash	20361	19:40:12 2023-12-20	<button>View Detail</button>
U000000005	5	Felic Cockshott	Cash	12119	19:40:12 2023-12-20	<button>View Detail</button>
U000000006	6	Camel Bruni	Momo	3260	19:40:12 2023-12-20	<button>View Detail</button>
U000000007	7	Amble Niese	Cash	3702	19:40:12 2023-12-20	<button>View Detail</button>
U000000008	8	Tore Puig	Momo	27034	19:40:12 2023-12-20	<button>View Detail</button>

- Admin, nhân viên và nha sĩ:** Khi nhấn “View detail”, người dùng có thể xem thông tin chi tiết hóa đơn.

The screenshot displays the System Admin Dashboard interface. On the left, a sidebar menu includes: Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main content area is divided into two sections: **Invoice Detail** and **Medicine Detail**.

**Invoice Detail:**

Select Treatment	Payment Id
1	Momo

Tooth Price (\$)	Medicine Price (\$)	Invoice Total (\$)
3746	16889	20635

Amount Paid (\$)	Change (\$)	Time
20635	0	20-12-2023 19:40:12

**Medicine Detail:**

MEDICINE NAME	UNIT PRICE (\$)	QUANTITY	TOTAL (\$)
Monobasic Sodium Phosphate	59	2	118
Jute	331	1	331
Sertapenem sodium	822	20	16440

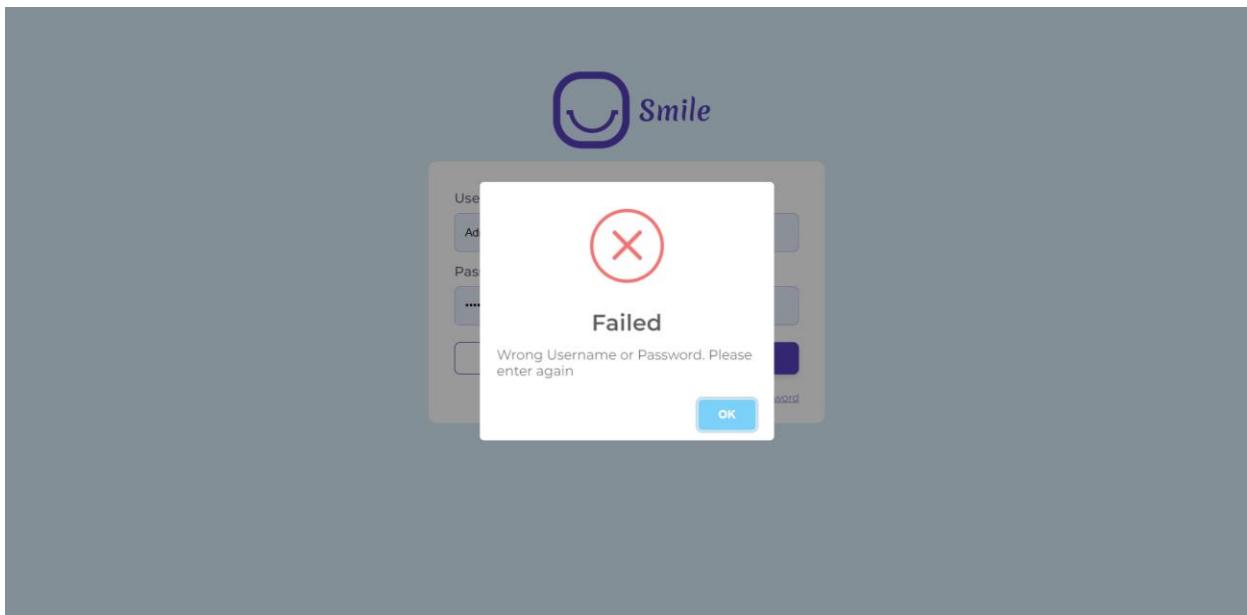
**Tooth Detail:**

TOOTH ID	SURFACE ID	PRICE (\$)
2	Mesial	600
3	Mesial	1315

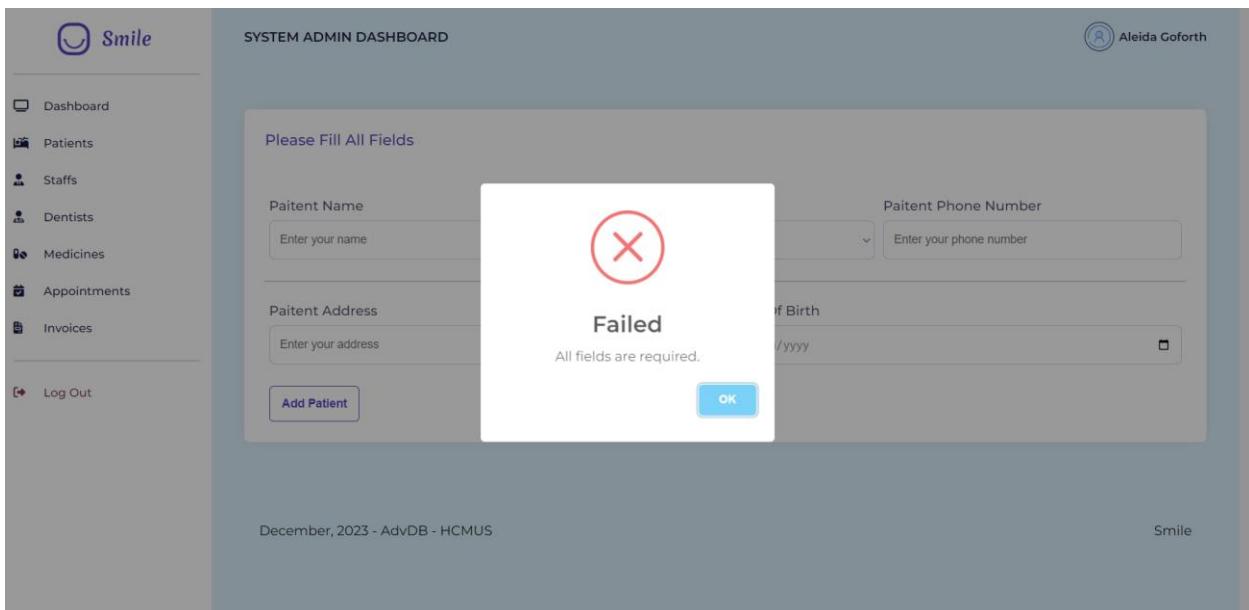
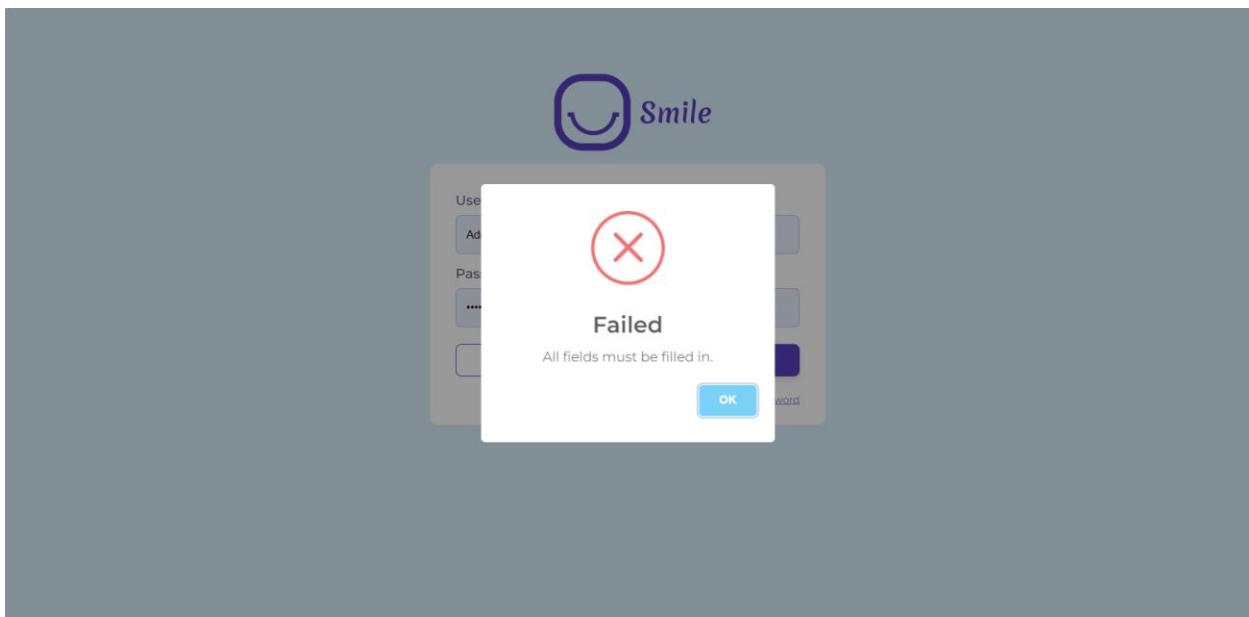
The screenshot shows the Smile dental management system. On the left is a sidebar with a logo, navigation links for Dashboard, Patients, Staffs, Dentists, Medicines, Appointments, Invoices, and Log Out. The main content area has two tables: 'Tooth Detail' and 'Treatment Detail'. The 'Tooth Detail' table has columns for TOOTH ID, SURFACE ID, and PRICE (\$). It contains two rows: one for tooth ID 2 with surface Mesial and price 600, and another for tooth ID 3 with surface Mesial and price 1315. The 'Treatment Detail' table has columns for TREATMENT ID and PRICE (\$). It contains two rows: one for treatment ID 12 with price 646, and another for treatment ID 26 with price 1185. At the bottom left is the text 'December, 2023 - AdvDB - HCMUS' and at the bottom right is the word 'Smile'.

## 2. Validation trong ứng dụng:

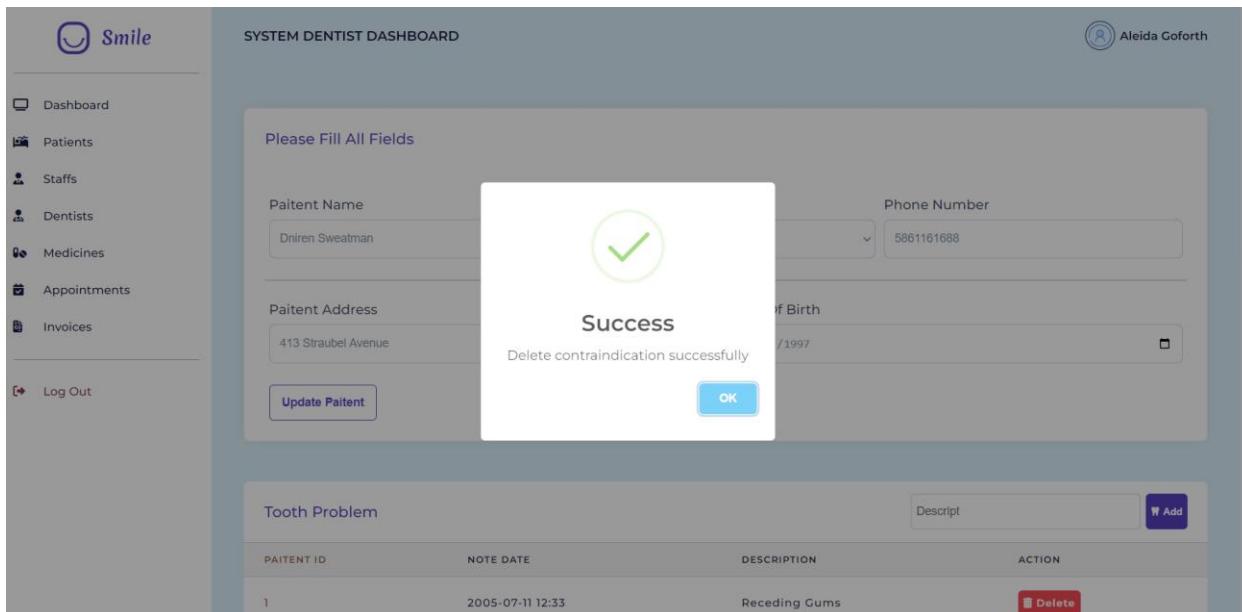
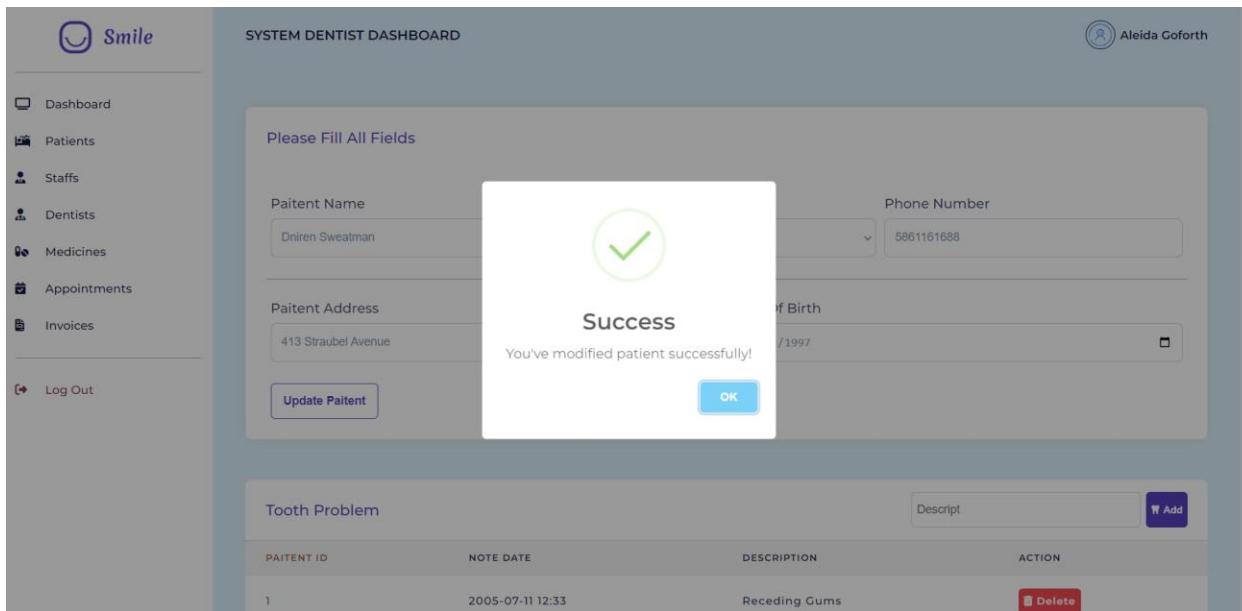
- Người dùng nhập sai thông tin.



- Yêu cầu người dùng nhập đầy đủ các thông tin.



- Người dùng thao tác thành công.



## IX. TÀI LIỆU THAM KHẢO

- [1] <https://guides.visual-paradigm.com/navigating-the-three-levels-of-database-design-conceptual-logical-and-physical/>
- [2] <https://viblo.asia/p/mot-so-cach-de-tang-toc-truy-van-sql-ORNZqgQ850n>
- [3] <https://niithanoi.edu.vn/huong-dan-5-buoc-phan-trang-pagivation-trong-lap-trinh-web-php.html>