

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG ĐIỆN - ĐIỆN TỬ



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài:

**THIẾT KẾ HỆ THỐNG QUẢN LÝ HỖ TRỢ BỆNH
NHÂN ĐIỀU TRỊ THẬN NHÂN TẠO**

Sinh viên thực hiện: ĐÌNH GIANG LONG

Lớp ĐTVT 03 – K63

Giảng viên hướng dẫn: TS. PHẠM THÀNH CÔNG

Hà Nội, 6-2023

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG ĐIỆN - ĐIỆN TỬ



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài:

**THIẾT KẾ HỆ THỐNG QUẢN LÝ HỖ TRỢ BỆNH
NHÂN ĐIỀU TRỊ THẬN NHÂN TẠO**

Sinh viên thực hiện: ĐINH GIANG LONG

Lớp ĐTVT 03 – K63

Giảng viên hướng dẫn: TS. PHẠM THÀNH CÔNG

Cán bộ phản biện:

Hà Nội, 6-2023

ĐÁNH GIÁ QUYỀN ĐỒ ÁN TỐT NGHIỆP

(Dùng cho giảng viên hướng dẫn)

Tên giảng viên đánh giá:TS. PHẠM THÀNH CÔNG.....

Họ và tên sinh viên:ĐINH GIANG LONG.....MSSV: 20182648.....

Tên đồ án:THIẾT KẾ HỆ THỐNG QUẢN LÝ HỖ TRỢ BỆNH NHÂN ĐIỀU TRỊ
.....BỆNH THẬN NHÂN TẠO

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)						
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án	1	2	3	4	5
2	Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế)	1	2	3	4	5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề	1	2	3	4	5
4	Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được	1	2	3	4	5
Có khả năng phân tích và đánh giá kết quả (15)						
5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống	1	2	3	4	5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng	1	2	3	4	5
7	Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai	1	2	3	4	5
Kỹ năng viết quyền đồ án (10)						
8	Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến; căn lề thống nhất, có dấu cách sau dấu chấm, dấu phẩy v.v.), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo	1	2	3	4	5

	và có trích dẫn đúng quy định					
9	Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.)	1	2	3	4	5
Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)						
10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/Đạt giải SVNCKH giải 3 cấp Viện trở lên/Có giải thưởng khoa học (quốc tế hoặc trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh, sáng chế	5				
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị SVNCKH nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành (VD: TI contest)	2				
10c	Không có thành tích về nghiên cứu khoa học	0				
Điểm tổng		/50				
Điểm tổng quy đổi về thang 10						

Nhận xét khác (về thái độ và tinh thần làm việc của sinh viên)

.....

.....

.....

.....

.....

.....

Ngày: ... / ... / 20...

Người nhận xét

(Ký và ghi rõ họ tên)

ĐÁNH GIÁ QUYỀN ĐỒ ÁN TỐT NGHIỆP

(Dùng cho cán bộ phản biện)

Giảng viên đánh giá:

Họ và tên sinh viên:ĐINH GIANG LONG.....MSSV: 20182648.....

Tên đồ án:THIẾT KẾ HỆ THỐNG QUẢN LÝ HỖ TRỢ BỆNH NHÂN ĐIỀU TRỊ
.....BỆNH THẬN NHÂN TẠO

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)					
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án	1	2	3	4 5
2	Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế)	1	2	3	4 5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề	1	2	3	4 5
4	Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được	1	2	3	4 5
Có khả năng phân tích và đánh giá kết quả (15)					
5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống	1	2	3	4 5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng	1	2	3	4 5
7	Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai	1	2	3	4 5
Kỹ năng viết quyền đồ án (10)					
8	Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến; căn lề thống nhất, có dấu cách sau dấu chấm, dấu phẩy v.v.), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo	1	2	3	4 5

	và có trích dẫn đúng quy định					
9	Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.)	1	2	3	4	5
Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)						
10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/Đạt giải SVNCKH giải 3 cấp Viện trở lên/Có giải thưởng khoa học (quốc tế hoặc trong nước) từ giải 3 trở lên/Có đăng ký bằng phát minh, sáng chế	5				
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị SVNCKH nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành (VD: TI contest)	2				
10c	Không có thành tích về nghiên cứu khoa học	0				
Điểm tổng		/50				
Điểm tổng quy đổi về thang 10						

Nhận xét khác của cán bộ phản biện

.....

.....

.....

.....

.....

.....

Ngày: ... / ... / 20...

Người nhận xét

(Ký và ghi rõ họ tên)

LỜI NÓI ĐẦU

Công tác quản lý thông tin các nhân của bệnh nhân cùng thông tin bệnh án, lịch sử các xét nghiệm ở bệnh viện hiện nay thường được tiến hành trên sổ sách, nội dung quản lý thông tin thường thiếu khoa học. Đặc biệt với bệnh nhân điều trị thân nhân tạo thì tần xuất chạy thận và xét nghiệm là rất thường xuyên khoảng 3-4 lần 1 tuần. Vì vậy với cách quản lý này không đáp ứng được nhu cầu cho bệnh nhân và các cán bộ y tế. Chính vì vậy, em đã lựa chọn đề tài: “ **Thiết kế hệ thống quản lý hỗ trợ bệnh nhân điều trị thận nhân tạo** ”, giúp cho quá trình quản lý được dễ dàng, chính xác hơn, hỗ trợ tốt cho bệnh viện, trung tâm y tế hay các phòng khám.

Trong quá trình thực hiện đồ án, em xin gửi lời cảm ơn chân thành tới TS. Phạm Thành Công và PGS TS. Vũ Duy Hải, Trường Điện tử - Viễn thông, Đại học Bách Khoa Hà Nội, đã hướng dẫn rất tận tình, nhiệt tình chỉ bảo và đưa ra hướng nghiên cứu tối ưu nhất để em có thể thực hiện được các yêu cầu của đề tài, từ đó có thể hoàn thiện đồ án một cách tốt nhất.

Trong thời gian thực hiện đồ án đã giúp em phát triển hơn rất nhiều không chỉ về kiến thức, trong việc tự mình giải quyết các vấn đề gặp phải từ giai đoạn tổng hợp yêu cầu đến lúc hoàn thành được sản phẩm cuối cùng. Đây cũng là học phần giúp em nhiều nhất trong suốt năm năm học đại học trước khi bắt đầu đi tìm một công việc cho bản thân sau khi ra trường. Việc thực hiện đồ án em không tránh khỏi những sai sót và hạn chế nhất định. Vì vậy, em rất mong nhận được sự góp ý và bổ sung của các thầy cô để đề tài được tối ưu và hoàn thiện nhất có thể.

Em xin chân thành cảm ơn!

LỜI CAM KẾT

Tôi là Đinh Giang Long, mã số sinh viên 20182648, sinh viên lớp Điện Tử Viễn Thông 03, khóa 63. Người hướng dẫn là TS. Phạm Thành Công. Tôi xin cam đoan toàn bộ nội dung được trình bày trong đồ án *Thiết kế hệ thống quản lý hỗ trợ bệnh nhân điều trị thận nhân tạo* là kết quả quá trình tìm hiểu và nghiên cứu của tôi. Các dữ liệu được nêu trong đồ án là hoàn toàn trung thực, phản ánh đúng kết quả thực tế. Mọi thông tin trích dẫn đều tuân thủ các quy định về sở hữu trí tuệ; các tài liệu tham khảo được liệt kê rõ ràng. Tôi xin chịu hoàn toàn trách nhiệm với những nội dung được viết trong đồ án này.

Hà Nội, ngày 06 tháng 07 năm 2023

Người cam đoan

Đinh Giang Long

MỤC LỤC

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT	i
DANH MỤC HÌNH VẼ	ii
DANH MỤC BẢNG BIỂU.....	iii
TÓM TẮT ĐỒ ÁN	iv
CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI.....	1
1.1 Tổng quan đề tài	1
1.1.1 Đặt vấn đề	1
1.1.2 Mục đích nghiên cứu.....	1
1.1.3 Phương pháp nghiên cứu đề tài đồ án	2
1.1.4 Quá trình thực hiện	2
1.2 Các công nghệ được sử dụng.....	2
1.2.1 Ngôn ngữ lập trình C#	2
1.2.2 SQL Server.....	3
1.2.3 Visustudio Code 2022 IDE	3
1.3 Yêu cầu về phần cứng.....	4
1.4 Yêu cầu về phần mềm.....	4
CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU	5
2.1 Khảo sát quy trình nghiệp vụ	5
2.2 Mô tả hệ thống phần mềm.....	7
2.3 Yêu cầu chức năng của hệ thống.....	8
2.3.1 Các yêu cầu chức năng của hệ thống	8
2.3.2 Các yêu cầu phi chức năng	9
2.3.2.1 Về hiệu năng	9
2.3.2.2 Về tính bảo mật, an toàn và khả năng phục hồi	9
2.3.2.3 Về tính bảo trì và giao diện	9
2.4 Xây dựng cơ sở dữ liệu	10
2.4.1 Các thực thể	10
2.4.2 Mối kết hợp của các thực thể	12
2.4.2.1 Mối quan hệ giữa Province - District -Ward	12
2.4.2.2 Mối quan hệ giữa User - Role, Province, District, Ward.....	13
2.4.2.3 Mối liên hệ giữa PatientDisease với Disease	14
2.4.2.4 Mối liên hệ giữa PatientSymptom với Symptom	14
2.4.2.5 Mối quan hệ giữa PatientDescription với Medicine.....	15
2.4.2.6 Mối quan hệ giữa thực thể PatientAppointment với Doctor, PatientRegistration	15

2.4.2.7	Mối quan hệ giữa thực thể <i>PatientAppointment</i> với <i>PatientDisease</i>	17
2.4.2.8	Mối quan hệ giữa thực thể <i>PatientAppointment</i> với <i>BioTest</i> , <i>UrineTest</i> , <i>BloodTest</i>	18
2.4.3	Thiết kế ERD (Entity Relationship Diagram).....	19
2.4.4	Cơ sở dữ liệu	20
2.4.4.1	Bảng <i>Role</i> (Vai trò).....	20
2.4.4.2	Bảng <i>User</i> (Tài khoản người dùng).....	20
2.4.4.3	Bảng <i>Doctor</i> (Bác sĩ).....	21
2.4.4.4	Bảng <i>Nurse</i> (Y tá).....	21
2.4.4.5	<i>PatientAppointment</i> (Cuộc hẹn khám bệnh của bệnh nhân).....	22
2.4.4.6	Bảng <i>Province</i> (Tỉnh thành)	22
2.4.4.7	Bảng <i>District</i> (Quận).....	23
2.4.4.8	Bảng <i>Ward</i> (Phường).....	23
2.4.4.9	Bảng <i>PatientSymptoms</i> (Triệu chứng của bệnh nhân).....	24
2.4.4.10	Bảng <i>Medicine</i> (Thuốc).....	24
2.4.4.11	Bảng <i>Disease</i> (Bệnh).....	25
2.4.4.12	Bảng <i>PatientRegistration</i> (Bệnh nhân đăng ký)	25
2.4.4.13	Bảng <i>PatientPrescription</i> (Đơn thuốc của bệnh nhân).....	25
2.4.4.14	Bảng <i>PatientDisease</i> (Kết quả khám bệnh của bệnh nhân).....	26
2.4.4.15	Bảng <i>UrineTest</i> (Kết quả kiểm tra Urine).....	26
2.4.4.16	Bảng <i>BloodTest</i> (Kết quả kiểm tra máu).....	27
2.5	Kết Luận	27
CHƯƠNG 3. THIẾT KẾ VÀ TRIỂN KHAI PHẦN MỀM		28
3.1	Thiết kế phần mềm	28
3.2	Mô hình chương trình tổng quát	28
3.1.1	Window Navigation View Diagram.....	30
3.1.2	Net Work Model Diagram.....	31
3.2	Xây dựng chương trình Main Winform Project	32
3.2.1	Thiết kế phân giao diện	32
3.2.1.1	Các màn hình cơ sở.....	32
3.2.1.2	Màn hình <i>SettingScreen</i>	34
3.2.1.3	Màn hình <i>LoginScreen</i>	35
3.2.1.4	Màn hình <i>HomeScreen</i>	36
3.2.1.5	Màn hình <i>StaffScreen</i>	37
3.2.1.6	Màn hình <i>MedicineScreen</i>	38
3.2.1.7	Màn hình <i>SysptomScreen</i>	39
3.2.1.8	Màn hình <i>BioTestScreen</i>	40
3.2.1.9	Màn hình <i>UrineTestScreen</i>	42
3.2.1.10	Màn hình <i>BloodTestScreen</i>	44

3.2.1.11 Màn hình HealthCheckScreen.....	45
3.2.1.12 Màn Hình PaymentScreen	48
3.2.2 Xây dựng lớp Log Contronller.....	49
3.2.2.1 Khái niệm chung.....	49
3.2.2.2 Xây dựng chương trình.....	49
3.2.3 Xây dựng các lớp cơ sở.....	50
3.2.3.1 Setting class	50
3.2.3.2 Login class.....	50
3.3 Xây dựng chương trình con Common Control UI Project	51
3.4 Xây dựng chương trình con Common Handle DB Project.....	52
3.5 Kiểm thử phần mềm.....	54
3.5.1 Kiểm tra chức năng	54
3.5.1.1 Chức năng Login	54
3.5.1.2 Chức năng lựa chọn tác vụ ở Giao diện HomScreen.....	54
3.5.1.3 Chức năng ADD	55
3.5.1.4 Chức năng EDIT.....	55
3.5.1.5 Chức năng DELETE.....	56
3.5.1.6 Hiển thị thông tin ở màn BioTesScreent	57
3.6 Kết Luận.....	58
CHƯƠNG 4. TỔNG KẾT VÀ PHÁT TRIỂN	59
4.1 Kết quả đạt được	59
4.2 Hạn chế của phần mềm.....	59
4.3 Hướng phát triển	59
KẾT LUẬN	60
TÀI LIỆU THAM KHẢO.....	61

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

STT	Kí hiệu	Thuật ngữ đầy đủ
1	DTTNT	Điều trị thận nhân tạo
2	CSDL	Cơ sở dữ liệu
3	CCCD	Căn cước công dân
4	API	Application Programing Interface
5	CPU	Central Processing Unit
6	XML	Extensible Markup Language
7	SQL	Structured Query Language
8	CSDLTT	Cơ sở dữ liệu tập trung
9	CRUD	Create, Read, Update, và Delete

DANH MỤC HÌNH VẼ

Hình 2.1 Hình khảo sát yêu cầu nghiệp vụ	5
Hình 2.2 Mô tả hệ thống phần mềm	7
Hình 2.3 Mối quan hệ giữa Province – District – Ward	12
Hình 2.4 Mối quan hệ giữa User – Role, Province, District, Ward	13
Hình 2.5 Mối quan hệ giữa PatientDisease với Disease	14
Hình 2.6 Mối quan hệ giữa PatientSymptom với Symptom.....	14
Hình 2.7 Mối quan hệ giữa PatientDescription với Medicine	15
Hình 2.8 Mối quan hệ giữa PatientAppointment với Doctor, PatientRegistration	16
Hình 2.9 Mối quan hệ giữa PatientAppointment với PatientDisease	17
Hình 2.10 Mối quan hệ giữa PatientAppointment với BioTest, BloodTest, UrineTest	18
Hình 2.11 Thiết kế ERD.....	19
Hình 3.1 Mô hình chương trình tổng quát	28
Hình 3.2 Window Negative Diagram	30
Hình 3.3 Net Work Model Diagram.....	31
Hình 3.4 Màn hình Sample	32
Hình 3.5 Sample2 Windows	33
Hình 3.6 Màn hình SettingScreen.....	34
Hình 3.7 Màn hình LoginScreen.....	35
Hình 3.8 Màn hình HomeScreen	36
Hình 3.9 Màn hình StaffScreen	37
Hình 3.10 Màn hình MedicineScreen	38
Hình 3.11 Màn hình SysptomScreen	39
Hình 3.12 Màn hình BioTestScreen.....	40
Hình 3.13 Màn hình UrineTestScreen	42
Hình 3.14 Màn hình BloodTestScreen.....	44
Hình 3.15 Màn hình HealthScreen.....	45
Hình 3.16 Màn hình PaymentScreen	48

DANH MỤC BẢNG BIỂU

Bảng 1.1 Yêu cầu phần cứng	4
Bảng 1.2 Yêu cầu phần mềm	4
Bảng 2.1 Cơ sở dữ liệu bảng Role(Vai trò).....	20
Bảng 2.2 Cơ sở dữ liệu bảng User (Tài khoản người dùng).....	20
Bảng 2.3 Cơ sở dữ liệu bảng Doctor (Bác sĩ).....	21
Bảng 2.4 Cơ sở dữ liệu bảng Nurse (Y tá).....	21
Bảng 2.6 Cơ sở dữ liệu bảng PartientAppointment (Cuộc hẹn khám bệnh của bệnh nhân).....	22
Bảng 2.7 Cơ sở dữ liệu bảng Provine (Tỉnh thành)	22
Bảng 2.8 Cơ sở dữ liệu bảng Distrisct (Quận).....	23
Bảng 2.9 Cơ sở dữ liệu bảng Ward (Phường).....	23
Bảng 2.10 Cơ sở dữ liệu bảng PatientSymptoms (Triệu chứng của bệnh nhân)	24
Bảng 2.11 Cơ sở dữ liệu bảng Medicine (Thuốc).....	24
Bảng 2.12 Cơ sở dữ liệu bảng Disease (Bệnh)	25
Bảng 2.13 Cơ sở dữ liệu bảng PatientRegistration (Bệnh nhân đăng ký)	25
Bảng 2.14 Cơ sở dữ liệu bảng PatientRescription (Đơn thuốc của bệnh nhân)	25
Bảng 2.15 Cơ sở dữ liệu bảng PatientDisease (Kết quả khám bệnh của bệnh nhân).....	26
Bảng 2.16 Cơ sở dữ liệu bảng UrineTest (Kết quả kiểm tra Urine)	26
Bảng 2.17 Cơ sở dữ liệu bảng BloodTest (Kết quả kiểm tra máu).....	27
Bảng 3.1 Các thành phần trên màn hình Sample	33
Bảng 3.2 Các thành phần trên màn hình SettingScreen.....	34
Bảng 3.3 Các thành phần trên màn hình LoginScreen	35
Bảng 3.4 Các thành phần trên màn hình MedicineScreen	38
Bảng 3.5 Các thành phần trên màn hình SysptomScreen	39
Bảng 3.6 Các thành phần trên màn hình BioTestScreen.....	40
Bảng 3.7 Các thành phần trên màn hình UrineTestScreen	42
Bảng 3.8 Các thành phần trên màn hình BloodTestScreen.....	44
Bảng 3.9 Phương thức chính trong Class CCUP	51
Bảng 3.11 Kiểm thử chức năng Login	54
Bảng 3.12 Kiểm thử chức năng lựa chọn tác vụ ở Giao diện HomeScreen	54
Bảng 3.13 Kiểm thử chức năng ADD.....	55
Bảng 3.14 Kiểm thử chức năng EDIT	55

Bảng 3.15 Kiểm thử chức năng DELETE	56
Bảng 3.16 Kiểm thử chức năng hiển thị trên màn hình BioTestScreen.....	57

TÓM TẮT ĐỒ ÁN

Với mong muốn được giúp đỡ cho việc khám chữa bệnh với tần suất cao, em quyết định lựa chọn xây dựng nên một phần mềm “**Hệ thống hỗ trợ bệnh nhân điều trị thân nhân tạo**”. Phần mềm có thể quản lý tốt thông tin bệnh nhân, thông tin xét nghiệm, điều trị cũng như đưa ra một số thống kê, biểu đồ một cách thuận tiện, dễ dàng và chính xác nhất để hỗ trợ cho việc khám chữa bệnh hiệu quả hơn.

Đồ án được chia thành 4 phần chính ứng với 4 chương. Cụ thể:

Chương 1: Trình bày về tổng quan đề tài, về yêu cầu, mục đích và ứng dụng của đề tài vào thực tiễn, đồng thời trình bày cơ sở lý thuyết phục vụ quá trình thiết kế và xây dựng hệ thống.

Chương 2: Trình bày các bước phân tích hệ thống, xây dựng cơ sở dữ liệu của hệ thống quản lý thông tin cho bệnh nhân.

Chương 3: Trình bày việc từng bước triển khai xây dựng chương trình từ phân chia hệ thống thành các project con để có thể sử dụng như các API, đến xây dựng các màn hình đủ để quản lý thông tin và hệ thống log cho quá trình chạy thực tế sau này.

Chương 4: Trình bày việc tổng kết và đưa ra hướng phát triển cho phần mềm.

Phần kết luận: Trình bày những kết quả đã thực hiện được và chưa thực hiện được và tổng kết lại đồ án đã thực hiện.

ABSTRACT

With the desire to help for medical examination and treatment with high frequency, I decided to choose to build a software "Patients support system for artificial body treatment". The software can well manage patient information, test information, treatment as well as provide some statistics and charts in the most convenient, easy and accurate way to support effective medical examination and treatment. more fruitful.

The project is divided into 4 main parts corresponding to 4 chapters. Specifically:

Chapter 1: Presents an overview of the topic, requirements, purposes and applications of the topic in practice, and presents the theoretical basis for the system design and construction process.

Chapter 2: Presenting the steps of systemanalysis, building a database of an information management system for patients.

Chapter 3: Presents the step-by-step implementation of building a program from dividing the system into sub-projects that can be used as APIs, to building screens sufficient to manage information and log systems for the process. the actual runner later.

Chapter 4: Presents the summary and gives development direction for the software.

Conclusion: Present the results that have been done and have not been implemented and summarize the project that has been implemented.

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI

Chương 1 đề cập tới việc giới thiệu tổng quan đề tài quản lý thông tin hỗ trợ bệnh nhân điều trị thân nhân tạo, đặt vấn đề, mô tả các phương pháp hiện có để giải quyết vấn đề, trình bày mục đích của đề án song song với việc giới hạn phạm vi của vấn đề mà đề án sẽ tập trung giải quyết, tóm tắt cấu trúc đề án và nội dung tương ứng của các phần sẽ lần lượt được trình bày ở các chương tiếp theo.

1.1 Tổng quan đề tài

1.1.1 Đặt vấn đề

Việc quản lý các thông tin liên quan đến việc điều trị bệnh nhân thân nhân tạo và kết quả xét nghiệm tại các cơ sở khám chữa bệnh luôn là điều vô cùng quan trọng khi mà thời gian gắn liền với sinh mạng, từ đó việc tổ chức thông tin một cách nhanh chóng, khoa học để đảm bảo hiệu quả khám chữa bệnh được chính xác và nhanh chóng là vô cùng cần thiết. Tuy nhiên, bởi lượng thông tin quá lớn do nhiều yếu tố, nên việc quản lý bằng các cách phương thức truyền thống đã không đáp ứng được nhu cầu thực tế. Khi thể mạnh của công nghệ thông tin càng chứng tỏ sự hiệu quả khi nó tham gia vào việc xử lý các vấn đề tại hầu hết mọi mặt trong cuộc sống chúng ta. Nó đã chứng minh đây chính là là cách tối ưu ở hiện tại để tiết kiệm thời gian và có thể xử lý chính xác nhất từng số liệu mà cách quản truyền thống có thể dễ mắc rất dễ nhiều sai sót. Vì vậy, điều cần thiết lúc này là chúng ta sẽ xây dựng một phần mềm quản lý thông tin ở các cơ sở khám chữa bệnh có bệnh nhân điều trị thân nhân tạo để có thể giúp ban quản lý các cơ sở khám chữa bệnh hoàn thành công việc quản lý dễ dàng hơn, tiết kiệm thời gian đạt được hiệu quả cao nhất có thể. Với mong muốn đáp ứng nhu cầu đó, nên em quyết định lựa chọn đề tài: **“Thiết kế hệ thống phần mềm hỗ trợ bệnh nhân điều trị thân nhân tạo”** để việc khám chữa bệnh được hiệu quả hơn, tiết kiệm được công sức, thời gian, tài nguyên cho các y bác sĩ trong công cuộc cứu người.

1.1.2 Mục đích nghiên cứu

Phần mềm “Hệ thống hỗ trợ bệnh nhân điều trị thân nhân tạo” được xây dựng với mục đích thực hiện việc xây dựng và phát triển một hệ thống quản lý các yêu cầu lưu trữ thông tin của bác sĩ, bệnh nhân, các thông tin xét nghiệm, thống kê từ những dữ liệu lịch sử trước cho các bệnh viện phòng khám, từ đó nâng cao hiệu quả làm việc cho các cán bộ y tế, y tá, bác sĩ nâng cao chất lượng khám chữa bệnh trong các bệnh viện.

Sau khi thực hiện, đề tài đặt ra những chức năng sau:

- Tìm hiểu và xây dựng một hệ thống toàn diện giúp người dùng, cụ thể là các cán bộ y tế, bác sĩ y tá có thể sử dụng dễ dàng việc quản lý các thông tin đến bệnh nhân

trong suốt quá trình điều trị, đem lại hiệu quả nhất định trong việc quản lý và vận hành các khoa thận trong bệnh viện, phòng khám.

- Xây dựng cấu trúc giao diện khoa học dễ dùng và dễ mở rộng
- Xây dựng được một cơ sở dữ liệu an toàn, bảo mật.

1.1.3 Phương pháp nghiên cứu đề tài đồ án

Trong đề tài sử dụng các phương pháp nghiên cứu:

- Phương pháp phân tích và tổng hợp lý thuyết: Nghiên cứu về các bài toán quản lý thông tin ở các cơ sở khám chữa bệnh đang được sử dụng hiện nay và trước đây. Bên cạnh đó tìm hiểu về cơ sở dữ liệu quan hệ. Từ đó đi phân tích và thiết kế mô hình cơ sở dữ liệu.
- Phương pháp thực nghiệm: Quy trình xét nghiệm và thông số xét nghiệm tại khoa thận nhân tạo để tìm ra những yêu cầu cụ thể cho hệ thống của mình. Từ đó để xây dựng giao diện và cấu trúc chương trình cho phù hợp và tiện dụng nhất.

1.1.4 Quá trình thực hiện

Trong quá trình thực hiện đề tài đã thực hiện thông qua các bước:

- Khảo sát về việc chạy thận của bệnh nhân trên các hội nhóm mạng xã hội để hình dung những khó khăn chung. Tìm hiểu quy trình làm việc tại các khoa điều trị bệnh thận để định hình về hệ thống phần mềm cần xây dựng.
- Thu thập các mẫu thông tin tại nơi khảo sát để sử dụng làm thông tin cho cơ sở dữ liệu của hệ thống
- Ứng dụng các kiến thức đã được học trong những năm đại học để thiết kế cơ sở dữ liệu.
- Nhận sự giúp đỡ và hướng dẫn tận tình từ thầy
- Tham khảo kiến thức từ những người thực tế, bạn bè, sách giáo khoa, các công cụ tìm kiếm.

1.2 Các công nghệ được sử dụng

1.2.1 Ngôn ngữ lập trình C#

Ngôn ngữ lập trình C# là ngôn ngữ lập trình bậc trung được phát triển vào năm 2000 bởi Microsoft. C# được xây dựng dựa trên ngôn ngữ C và C++ nhưng dễ sử dụng hơn nhờ những bộ thư viện rộng lớn để thực hiện các các vụ khác nhau.

Tại sao nên dùng ngôn ngữ C#?

- Là một trong số những ngôn ngữ thuần hướng đối tượng.
- Chuyên sử dụng để lập trình cho windows.

- Thiết kế winform cực tốt, đơn giản và dễ hiểu.
- Khả năng tương tác với Database dễ dàng hơn rất nhiều.
- Được window hỗ trợ đầy đủ các control.
- Thư viện .NET nhẹ, dễ cài đặt và được miễn phí.
- Ngôn ngữ mã nguồn mở.
- Code/Build trên Visual Studio, một IDE tiện lợi, mạnh mẽ của Microsoft.

1.2.2 SQL Server

SQL Server còn có tên gọi khác là Microsoft SQL Server (viết tắt là MS SQL Server) một phần mềm được Microsoft phát triển nhằm mục đích lưu trữ các dữ liệu dựa trên chuẩn RDBMS (**Relational Database Management System**).

Hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System) sẽ sử dụng câu lệnh SQL nhằm trao đổi dữ liệu giữa máy Client và các máy đã cài ứng dụng SQL Server.

SQL Server sẽ cung cấp đầy đủ các công cụ phục vụ cho việc quản lý, từ giao diện GUI cho đến các ngôn ngữ dùng để truy vấn SQL. Ưu điểm nổi bật của SQL Server là chúng có thể kết hợp với nhiều nền tảng như ASP.NET hoặc C#.

Tại sao nên dùng SQL Server?

- Cho phép tạo nhiều cơ sở dữ liệu.
- Miễn phí.
- Duy trì lưu trữ bền vững.
- Bảo mật cao.
- SQL Server sẽ cung cấp đầy đủ các công cụ phục vụ cho việc quản lý, từ giao diện GUI cho đến các ngôn ngữ dùng để truy vấn SQL. Ưu điểm nổi bật của SQL Server là chúng có thể kết hợp với nhiều nền tảng như ASP.NET hoặc C#.

1.2.3 Visustudio Code 2022 IDE

Visual Studio 2022 là phiên bản mới nhất của phần mềm lập trình được phát triển bởi Microsoft và có sẵn để tải xuống từ trang web chính thức của Microsoft. Nó cung cấp hai bản phát triển khác nhau nhưng tính năng của 2 phiên bản là hoàn toàn giống nhau và không giới hạn về việc phát triển dự án này.

Tại sao nên dùng Visustudio Code 2022?

- Dễ tiếp cận
- Miễn phí
- Hỗ trợ nhiều ngôn ngữ lập trình và nền tảng

- IDE 64 bit: Visual Studio 2022 là IDE 64 bit có thể xử lý ngay cả những dự án lớn nhất.
- Tích hợp với nhiều tính năng cao cấp, mang lại tất cả những công cụ rất cần thiết để tạo ứng dụng.

1.3 Yêu cầu về phần cứng

Yêu cầu về phần cứng mà tôi dùng để phát triển hệ thống này được đưa ra như bên dưới, để ít xảy ra vấn đề thì phần cứng có thể như vậy hoặc cao hơn.

Bảng 1.1 Yêu cầu phần cứng

STT	Phần cứng	Yêu cầu tối thiểu	Chú thích
1	Processor	Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz	
2	Memory	8.00 GB RAM	

1.4 Yêu cầu về phần mềm

Yêu cầu về phần mềm tối thiểu được đề xuất để phát triển hệ thống này được liệt kê như bên dưới:

Bảng 1.2 Yêu cầu phần mềm

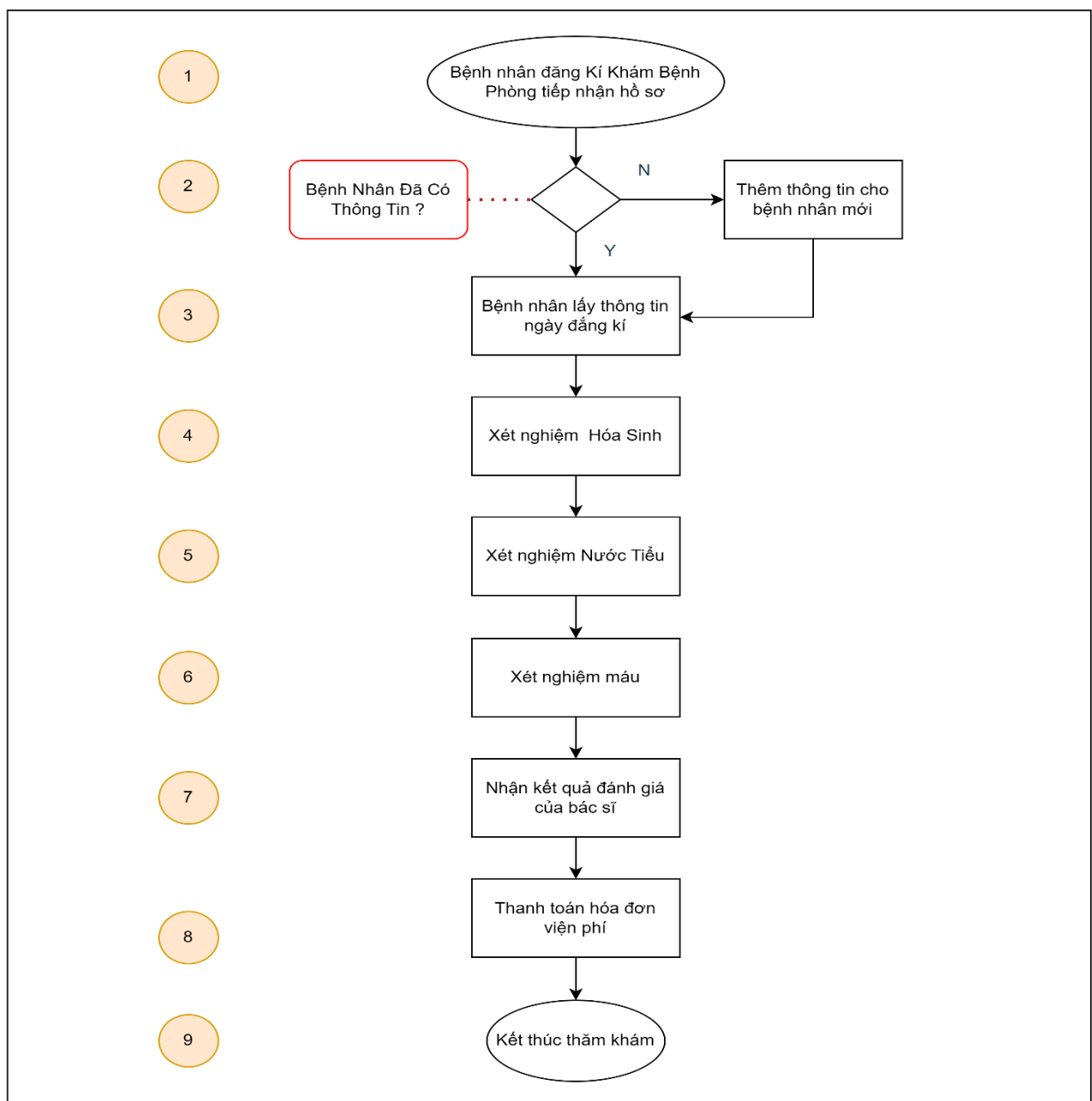
STT	Phần Mềm	Yêu cầu tối thiểu	Chú thích
1	Operating System (OS)	Hầu hết các hệ điều hành Window 10 - 64 bit.	

CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

Qua quá trình tìm hiểu cơ sở lý thuyết, ở chương 2 nghiên cứu và phân tích yêu cầu đề tài, tiến thành phân tích kỹ hệ thống và đưa ra thiết kế các khối chức năng, thiết kế xây dựng cơ sở dữ liệu cho hệ thống.

2.1 Khảo sát quy trình nghiệp vụ

Trước khi xây dựng hệ thống phần mềm quản lý thông tin bệnh nhân điều trị thận nhân tạo, đã tiến hành khảo sát về nghiệp vụ và quy trình xử lý nghiệp vụ liên quan. Dưới đây là mô tả phân tích quy trình lần thăm khám của bệnh nhân đi điều trị thận:



Hình 2.1 Hình khảo sát yêu cầu nghiệp vụ

Dựa vào quá trình của một lần thăm khám của bệnh nhân điều trị thân và bên cạnh đó là phân tích các khía cạnh của hệ thống để đáp ứng thì phần mềm có các chức năng chính như sau:

1) Đây là quy trình được thực hiện tại phòng tiếp nhận hồ sơ

Trong quy trình này, người dùng (nhân viên tiếp nhận) sẽ truy cập vào màn hình nhập thông tin bệnh nhân đến thăm khám. Ở đây, họ sẽ search theo số điện thoại và số CCCD(Căn cước công dân) để xác định bệnh nhân đã có thông tin trong hệ thống hay chưa.

2) Đây là quy trình được thực hiện tại phòng tiếp nhận hồ sơ

Trong quy trình này, sau khi kiểm tra sự tồn tại thông tin của bệnh nhân trong CSDL là chưa có. Nhân viên sẽ tiến hành nhập mới thông tin cho bệnh nhân những thông tin cơ bản như tên, tuổi, quê quán, giới tính... Nếu đã có, hệ thống sẽ chuyển bước tiếp theo, không cần nhập lại thông tin.

3) Đây là quy trình được thực hiện tại phòng tiếp nhận hồ sơ

Trong quy trình này, sau khi thông tin bệnh nhân đã được xác nhận và kiểm tra, nhân viên sẽ chuyển sang màn hình lên lịch khám cho người bệnh có thể khám ở các bước tiếp theo. Ở đây có thể thêm sửa xóa các thông tin như: Ngày thăm khám, người giám hộ,..

4) Đây là quy trình được thực hiện tại phòng Xét nghiệm hóa sinh

Sau khi bệnh nhân đã đăng kí lịch khám, tại phòng xét nghiệm hóa sinh, nhân viên sẽ sử dụng màn hình để lưu thông tin xét nghiệm hóa sinh cho bệnh nhân. Thông tin này bao gồm các chỉ số hóa sinh cần được kiểm tra, kết quả xét nghiệm, và các thông tin liên quan khác nếu cần thiết.

5) Đây là quy trình được thực hiện tại phòng Xét nghiệm nước tiểu

Tương tự như quy trình xét nghiệm hóa sinh, tại phòng xét nghiệm nước tiểu, nhân viên sẽ sử dụng màn hình xét nghiệm nước tiểu để lưu lại chỉ số xét nghiệm của xét nghiệm nước tiểu. Các thông tin này sẽ được lưu trữ và gắn liền với hồ sơ lượt đăng kí khám bệnh hiện tại của bệnh nhân.

6) Đây là quy trình được thực hiện tại phòng Xét nghiệm máu

Tương tự như quy trình xét nghiệm nước tiểu, tại phòng xét nghiệm máu, nhân viên sẽ sử dụng màn hình xét nghiệm máu để lưu lại chỉ số xét nghiệm máu. Các thông tin này sẽ được lưu trữ và gắn liền với hồ sơ lượt đăng kí khám bệnh hiện tại của bệnh nhân.

7) Đây là quy trình được thực hiện tại phòng Nhận kết quả xét nghiệm

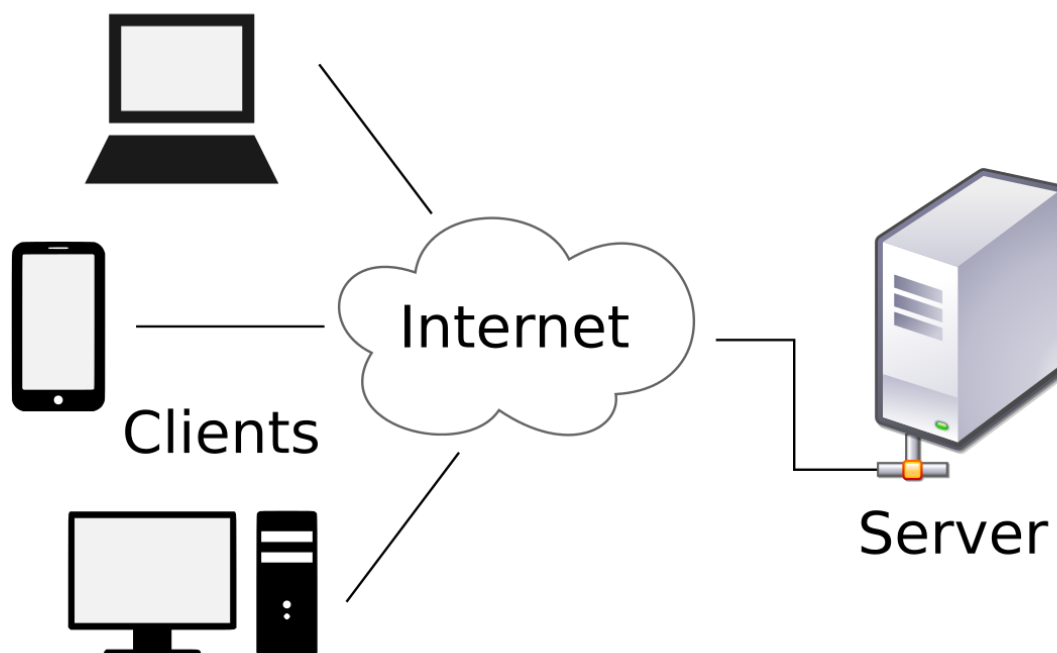
Sau khi có đủ thông tin xét nghiệm từ các phòng xét nghiệm, bác sĩ sẽ truy cập vào màn hình kết quả xét nghiệm và các màn hình thống kê lịch sử thăm khám của bệnh nhân để tham khảo. Tại đây, họ có thể kiểm tra và đánh giá kết quả xét nghiệm của bệnh nhân một cách dễ dàng hơn. Dựa trên kết quả này, bác sĩ có thể đưa ra đánh giá và quyết định việc điều trị, cũng như kê đơn thuốc cho bệnh nhân.

8) Đây là quy trình được thực hiện tại phòng Làm thủ tục ra viện

Cuối cùng, trong quy trình xét nghiệm của bệnh nhân, tại phòng làm thủ tục ra viện, nhân viên sẽ thực hiện các bước để hoàn tất việc ra viện cho bệnh nhân. Họ sẽ tạo hóa đơn và bao gồm tất cả các chi phí về xét nghiệm thuốc hay các chi phí phát sinh khác nếu có.

2.2 Mô tả hệ thống phần mềm

Hệ thống quản lý hỗ trợ bệnh nhân điều trị sẽ sử dụng mô hình Client Server. Là mô hình mạng gồm có 2 thành phần chính đó là máy khách (client) và máy chủ (server). Server chính là nơi giúp lưu trữ tài nguyên cũng như thực hiện các dịch vụ theo đúng như yêu cầu của client. Ngược lại, Client bao gồm máy tính cũng như các loại thiết bị điện tử nói chung sẽ tiến hành gửi yêu cầu đến server thông qua kết nối bằng địa chỉ IP và Port của Server.



Hình 2.2 Mô tả hệ thống phần mềm

Với tính chất và quy mô của phần mềm quản lý đặc thù trong bệnh viện thì mô hình Client-Server mang lại rất nhiều lợi thế vượt trội như:

- Tính tập trung thông tin: Vì toàn bộ thông tin về bệnh nhân đều tập trung vào một cơ sở duy nhất nên những thông tin quan trọng về lịch sử bệnh án, đơn thuốc, ... đảm bảo được tính toàn vẹn và nhất quán.
- Khả năng bảo mật: Có thể đưa các phương pháp bảo mật cao vào hệ thống, đảm bảo các thông tin quan trọng của bệnh nhân chỉ được truy cập khi được cấp quyền.
- Khả năng mở rộng và sao lưu dữ liệu: Với mô hình Client-Server cũng cho phép thực hiện một cách dễ dàng.

Bên cạnh những điểm lợi đó thì mô hình cũng mang những nhược điểm, mà nhược điểm lớn nhất chính là khi Server xảy ra sự cố thì toàn bộ hệ thống sẽ bị ảnh hưởng theo. Về chi phí để xây dựng vào bảo trì cũng thường là rất cao.

2.3 Yêu cầu chức năng của hệ thống

2.3.1 Các yêu cầu chức năng của hệ thống

- 1) Hệ thống hiện tại sẽ phân cấp với 3 Role: Admin, Bác Sĩ, Y Tá.
- 2) Quản lý tài khoản người dùng: Admin có quyền cấp tài khoản để đăng nhập vào hệ thống với các chức năng được sử dụng dựa theo Role của người dùng đó. Role: Admin.
- 3) Quản lý thông tin Người Bệnh: Cho phép CRUD (Create, Read, Update, và Delete) với các thông tin của bệnh nhân đến thăm khám. Thông tin người bệnh có tính duy nhất dựa vào số điện thoại và số CCCD. Role: Bác Sĩ, Y Tá.
- 4) Quản lý thông tin Bác Sĩ, Y Tá: Cho phép CRUD các thông tin của Bác Sĩ và Y Tá. Role: Admin.
- 5) Quản lý thông tin Thuốc: Cho phép CRUD các thông tin về các thuốc đang được sử dụng bao gồm tên và loại thuốc. Role: Admin.
- 6) Quản lý thông tin về các triệu chứng thường gặp: Cho phép CRUD các thông tin về các triệu chứng thường gặp của bệnh nhân. Role: Admin, Bác Sĩ.
- 7) Quản lý thông tin xét nghiệm Hóa Sinh: Cho phép CRUD các thông tin về các thông số xét nghiệm Hóa Sinh của bệnh nhân. Role: Admin, Bác Sĩ, Y tá.
- 8) Quản lý thông tin xét nghiệm Nước Tiểu: Cho phép CRUD các thông tin về các thông số xét nghiệm Nước Tiểu của bệnh nhân. Role: Admin, Bác Sĩ, Y tá.
- 9) Quản lý thông tin xét nghiệm Máu: Cho phép CRUD các thông tin về các thông số xét nghiệm Máu của bệnh nhân. Role: Admin, Bác Sĩ, Y tá.

- 10) Quản lý kết quả đánh giá bệnh và kê đơn thuốc: Cho phép đưa ra đánh giá bệnh án, lựa chọn đơn thuốc, liều lượng, thời gian uống cho từng loại thuốc. Role: Admin, Bác Sĩ.
- 11) Biểu đồ thể hiện quá trình điều trị của mỗi bệnh nhân: Từ lịch sử khám bệnh của mỗi bệnh nhân, hệ thống vẽ biểu đồ thể hiện một cách trực quan sự thay đổi các thông số theo mỗi lần để hỗ trợ cho các bác sĩ và y tế trong việc điều trị. Role: Admin, Bác Sĩ, Y Tá có thể xem.

2.3.2 Các yêu cầu phi chức năng

2.3.2.1 Về hiệu năng

Về mặt tốc độ xử lý, hệ thống cần đảm bảo có tốc độ nhanh, thời gian xử lý yêu cầu nhanh chóng, không gây ra sự khó chịu cho người sử dụng. Trong trường hợp có nhiều người cùng truy cập tại cùng một thời điểm, tốc độ của hệ thống vẫn đạt ở mức chấp nhận được.

Về mặt độ trễ, yêu cầu hệ thống cần có độ trễ thấp.

Về số lượng người dùng, yêu cầu hệ thống cần đáp ứng được số lượng người dùng.

2.3.2.2 Về tính bảo mật, an toàn và khả năng phục hồi

Hệ thống cần phải đảm bảo yếu tố bảo mật tuyệt đối mọi thông tin của người sử dụng.

Có cách thức truy nhập an toàn bằng mật khẩu. Kiểm soát người dùng và phân quyền chặt chẽ, chính xác. Cơ sở dữ liệu được sao lưu nhằm phòng tránh các trường hợp bị sai hoặc mất dữ liệu.

Trong các trường hợp bất ngờ hay các sự cố không mong muốn (ví dụ trong trường hợp mất điện, mất kết nối), hệ thống cần có các cơ chế bảo đảm an toàn cho dữ liệu, tránh sai sót thông tin.

Về khả năng phục hồi, hệ thống cần đảm bảo việc dễ phục hồi và sửa chữa dữ liệu nếu có sự cố xảy ra.

Vì hệ thống được thao tác trên môi trường internet nên có thể có rất nhiều vấn đề liên quan đến bảo mật hay sự cố do virus. Do vậy, các yếu tố liên quan đến bảo mật dữ liệu cần phải được coi trọng hơn bao giờ hết.

2.3.2.3 Về tính bảo trì và giao diện

Các chức năng, giao diện đòi hỏi cần phải dễ bảo trì, sửa đổi, bổ sung.

Đối với người dùng, hệ thống cần có giao diện trực quan, thân thiện, dễ sử dụng, dễ thao tác, dễ sử dụng và quản lý không gây ra sự khó chịu cho người sử dụng. Trong trường hợp có nhiều người cùng truy cập tại cùng một thời điểm, tốc độ của hệ thống vẫn đạt ở mức chấp nhận được.

Về mặt độ trễ, yêu cầu hệ thống cần có độ trễ thấp.

Về số lượng người dùng, yêu cầu hệ thống cần đáp ứng được số lượng người dùng.

2.4 Xây dựng cơ sở dữ liệu

Qua quá trình tìm hiểu cơ sở lý thuyết, nghiên cứu và phân tích yêu cầu đề tài, tiến thành phân tích kỹ hệ thống và đưa ra thiết kế các khối chức năng, thiết kế xây dựng cơ sở dữ liệu cho hệ thống.

2.4.1 Các thực thể

1) Thực thể Role (Quyền hạn)

Là một bảng lưu trữ thông tin về quyền hạn của User - Người dùng của hệ thống. Khi một người dùng được gán một quyền hạn, người dùng đó sẽ có quyền truy cập vào các đối tượng hệ thống có quyền truy cập.

2) Thực thể User (Tài khoản Người dùng)

Là một bảng lưu trữ thông tin về người dùng của hệ thống. Khi một người dùng được gán một quyền hạn, người dùng đó sẽ có quyền truy cập vào các đối tượng hệ thống có quyền truy cập.

3) Thực thể Doctor (Bác sĩ)

Là một bảng lưu trữ thông tin về bác sĩ của hệ thống. Mỗi bác sĩ có thể được gán cho nhiều bệnh nhân. Khi một bác sĩ được gán cho một bệnh nhân, bệnh nhân đó sẽ được bác sĩ đó điều trị.

4) Thực thể Nurse (Y tá)

Là một bảng lưu trữ thông tin về y tá của hệ thống. Mỗi y tá có thể được gán cho nhiều bệnh nhân. Khi một y tá được gán cho một bệnh nhân, bệnh nhân đó sẽ được y tá đó chăm sóc.

5) Thực thể Patient (Bệnh nhân)

Là một bảng lưu trữ thông tin về bệnh nhân của hệ thống. Bệnh nhân là người sử dụng dịch vụ chăm sóc sức khỏe của hệ thống. Được điều trị bởi các bác sĩ, y tá. Mỗi bệnh nhân có thể có nhiều hồ sơ bệnh án. Khi một bệnh nhân có một hồ sơ bệnh án, bệnh án đó sẽ lưu trữ tất cả thông tin về bệnh nhân, bao gồm lịch sử bệnh, chẩn đoán, phương pháp điều trị và các thông tin khác.

6) Thực thể PatientAppointment (đăng kí khám bệnh của bệnh nhân)

Là một bảng lưu trữ thông tin các cuộc hẹn giữa bệnh nhân và bác sĩ. Bệnh nhân là người sử dụng dịch vụ chăm sóc sức khỏe của hệ thống. Được điều trị bởi các bác sĩ, y tá.

Mỗi bệnh nhân có thể có nhiều hồ sơ bệnh án. Khi một bệnh nhân có một hồ sơ bệnh án, bệnh án đó sẽ lưu trữ tất cả thông tin về bệnh nhân, bao gồm lịch sử bệnh, chẩn đoán, phương pháp điều trị và các thông tin khác.

7) Thực thể Province (Tỉnh thành)

Là một bảng lưu trữ thông tin về 1 tỉnh.

8) Thực thể District (Quận)

Là một bảng lưu trữ thông tin về 1 quận. Quận là một đơn vị hành chính cấp dưới của tỉnh. Trong 1 tỉnh có thể có nhiều quận.

9) Thực thể Ward (Phường)

Là một bảng lưu trữ thông tin về 1 phường. Phường là một đơn vị hành chính cấp dưới của quận. Là đơn vị hành chính nhỏ nhất ở Việt Nam. Trong 1 tỉnh có thể có nhiều quận.

10) Thực thể PatientSymptoms (Triệu chứng của bệnh nhân)

Là một bảng lưu trữ thông tin về triệu chứng của bệnh nhân. Bác sĩ có thể sử dụng thông tin về triệu chứng của bệnh nhân để đưa ra chẩn đoán, phương pháp điều trị và sử dụng thuốc. Triệu chứng là một phần quan trọng trong chẩn đoán bệnh.

11) Thực thể Medicine (Thuốc)

Là một bảng lưu trữ thông tin về các loại thuốc. Bác sĩ có thể sử dụng thông tin về thuốc để kê đơn cho bệnh nhân. Thuốc là một phần quan trọng trong nghiệp vụ khám chữa bệnh.

12) Thực thể Disease (Bệnh)

Chứa thông tin về các chuẩn đoán bệnh cụ thể cho các bệnh nhân.

13) Thực thể PatientRescription (Đơn thuốc của bệnh nhân)

Là một bảng lưu trữ trong cơ sở dữ liệu lưu trữ thông tin về đơn thuốc đã được cấp cho bệnh nhân, cung cấp thông tin về các loại thuốc mà bệnh nhân đang dùng và giúp đảm bảo rằng họ đang dùng đúng cách.

14) Thực thể PatientDisease (Bản kết quả các loại bệnh của bệnh nhân)

Là một thực thể trong cơ sở dữ liệu lưu trữ thông tin về các bệnh mà bệnh nhân mắc phải, cung cấp thông tin về các bệnh mà bệnh nhân mắc phải và giúp đảm bảo rằng họ đang được điều trị đúng cách.

15) Thực thể UrineTest (Kết quả xét nghiệm nước tiểu)

Là một thực thể trong cơ sở dữ liệu lưu trữ thông tin về các xét nghiệm nước tiểu. cung cấp thông tin về các xét nghiệm nước tiểu của bệnh nhân và giúp đảm bảo rằng họ đang được điều trị đúng cách.

16) Thực thể BloodTest (Kết quả xét nghiệm máu)

Là một thực thể trong cơ sở dữ liệu lưu trữ thông tin về các xét nghiệm máu, cung cấp thông tin về các xét nghiệm máu của bệnh nhân và giúp đảm bảo rằng họ đang được điều trị đúng cách.

17) Thực thể BloodTest (Kết quả xét nghiệm hóa sinh)

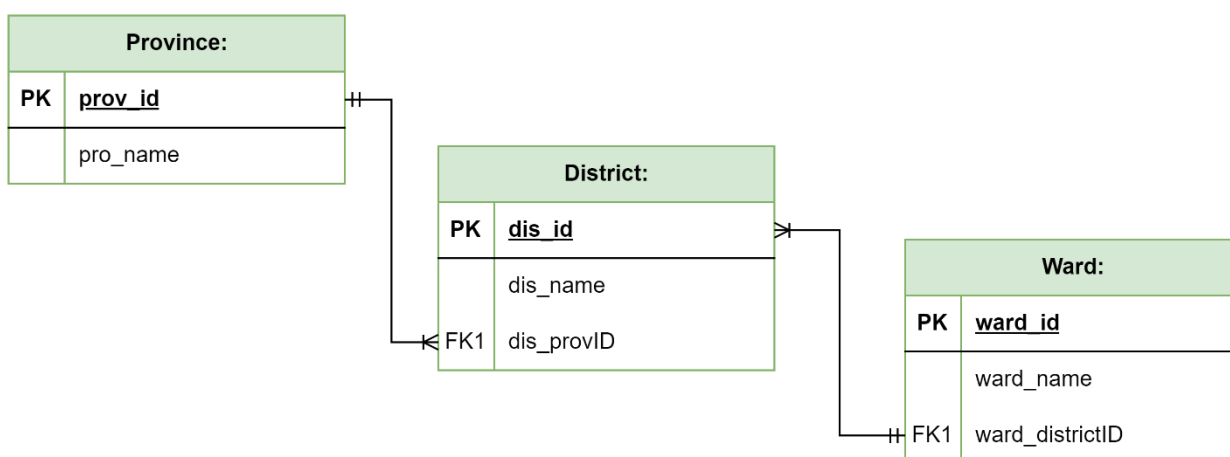
Là một thực thể trong cơ sở dữ liệu lưu trữ thông tin về các xét nghiệm hóa sinh, cung cấp thông tin về các xét nghiệm hóa sinh của bệnh nhân và giúp đảm bảo rằng họ đang được điều trị đúng cách.

18) Thực thể HealthCheckup

Là một thực thể chứa các thông tin đánh giá của bác sĩ và thông tin về chi phí mà bệnh nhân phải thanh toán trước khi làm thủ tục xuất viện

2.4.2 Mối kết hợp của các thực thể

2.4.2.1 Mối quan hệ giữa Province - District - Ward



Hình 2.3 Mối quan hệ giữa Province – Distret – Ward

Mối quan hệ giữa Province và District:

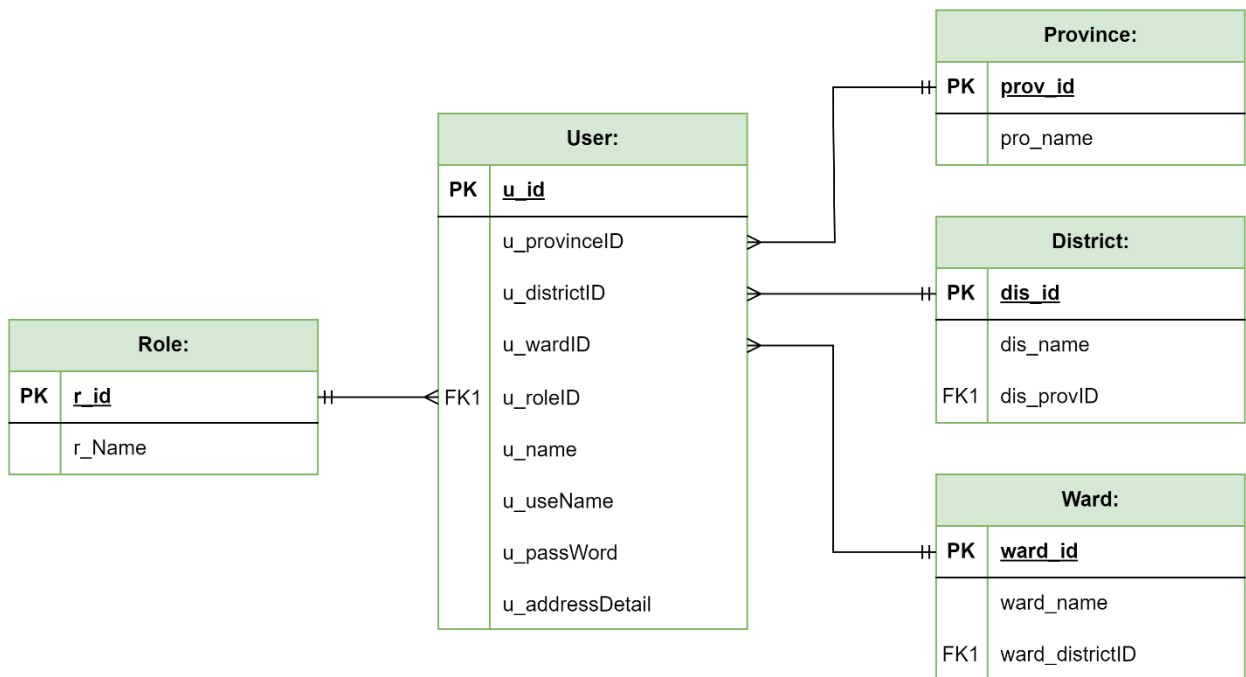
- Một Province (tỉnh/thành phố) có thể có nhiều District (huyện/quận).
- Mỗi District thuộc về một và chỉ một Province.
- Biểu thị: Province (1) ---- (N) District

Mối quan hệ giữa District và Ward:

- Một District (huyện/quận) có thể có nhiều Ward (phường/xã).
- Mỗi Ward thuộc về một và chỉ một District.
- Biểu thị: District (1) ----> (N) Ward

Dữ liệu về thông tin các tỉnh/thành phố, quận/huyện và phường/xã của Việt Nam sẽ được thêm vào database phục vụ cho việc lưu trữ thông tin về quê quán cho các thực thể khác.

2.4.2.2 *Mối quan hệ giữa User - Role, Province, District, Ward*



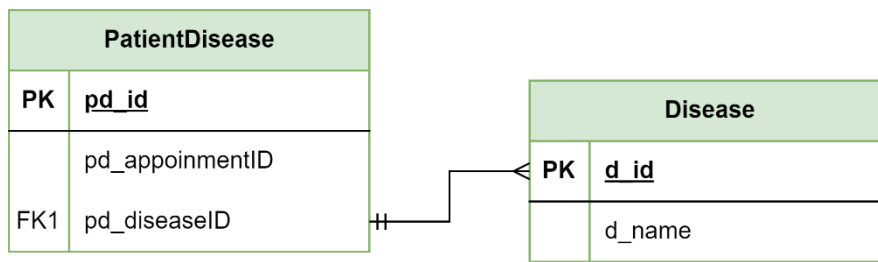
Hình 2.4 Mối quan hệ giữa User – Role, Province, District, Ward

Mối quan hệ giữa: User với Role:

- Một User chỉ thuộc về một Role
- Một Role có thể có ở nhiều User
- Biểu thị: Role (1) ---- (N) User

Tương tự như vậy mối quan hệ giữa: User với Province, District, Ward là tương tự như với mối quan hệ giữa như bên trên giữa User và Role.

2.4.2.3 Mối liên hệ giữa PatientDisease với Disease

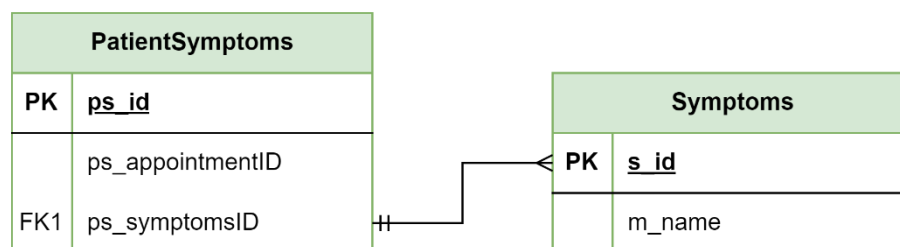


Hình 2.5 Mối quan hệ giữa PatientDisease với Disease

Mối quan hệ giữa PatientDisease với Disease:

- Một Disease chỉ thuộc về một PatientDisease
- Một PatientDisease có thể có ở nhiều Disease
- Biểu thị: PatientDisease (1) ---- (N) Disease

2.4.2.4 Mối liên hệ giữa PatientSymptom với Symptom

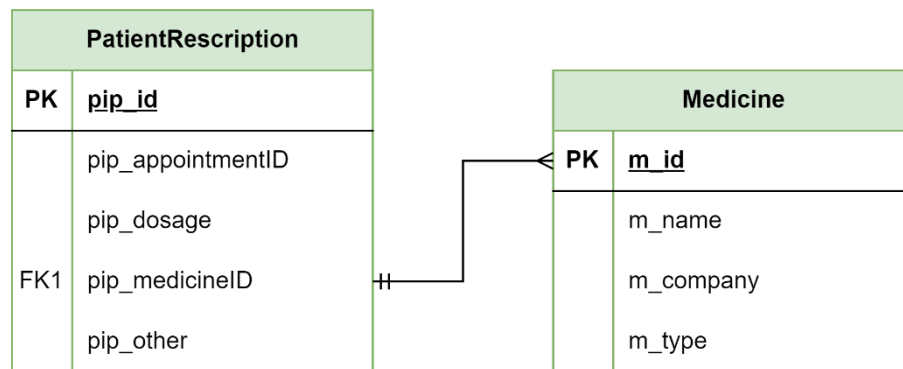


Hình 2.6 Mối quan hệ giữa PatientSymptom với Symptom

Mối quan hệ giữa PatientSymptom với Symptom:

- Một Symptom chỉ thuộc về một PatientSymptom
- Một PatientSymptom có thể có ở nhiều Symptom
- Biểu thị: PatientSymptom (1) ---- (N) Symptom

2.4.2.5 Mối quan hệ giữa PatientDescription với Medicine

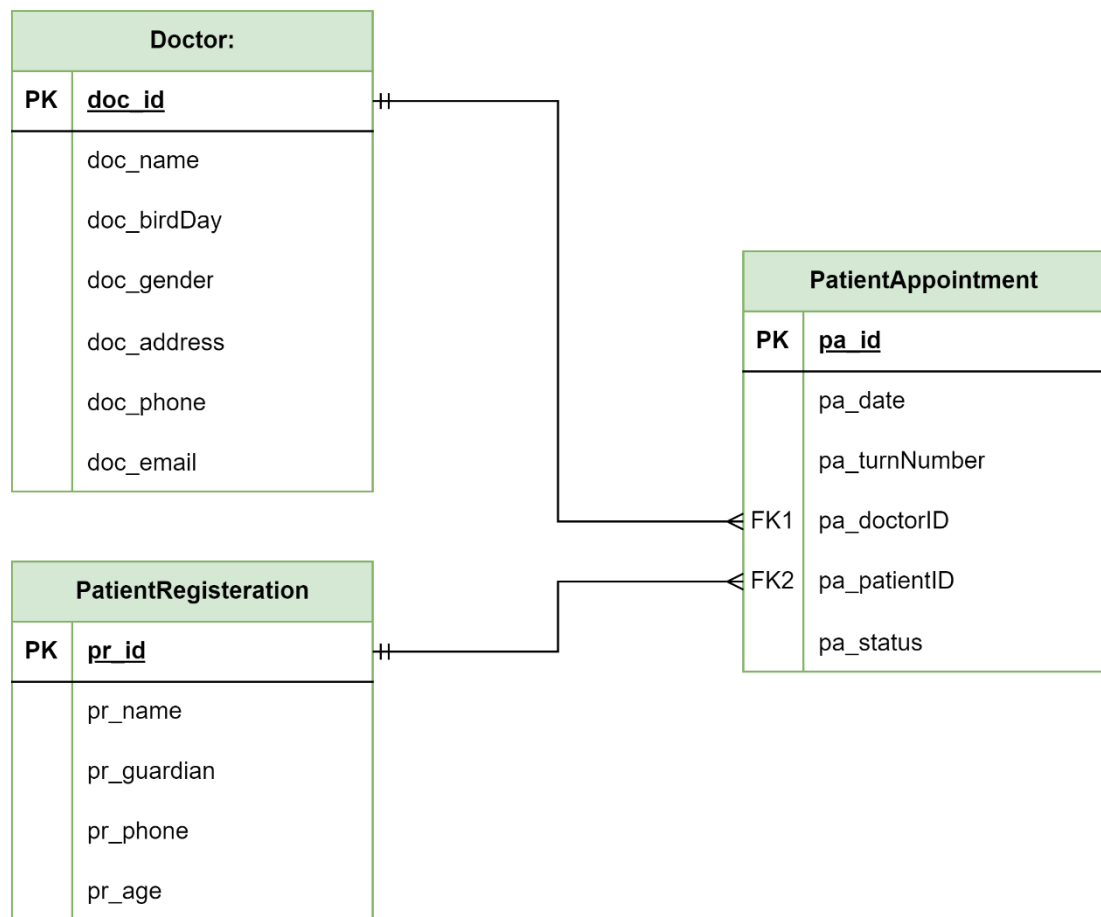


Hình 2.7 Mối quan hệ giữa PatientDescription với Medicine

Mối quan hệ giữa PatientDescription với Medicine:

- Một PatientDescription chỉ chứa một Medicine
- Một Medicine có thể có ở nhiều PatientDescription
- Biểu thị: PatientDescription (1) ---- (N) Medicine

2.4.2.6 Mối quan hệ giữa thực thể PatientAppointment với Doctor, PatientRegistration



Hình 2.8 Mối quan hệ giữa PatientAppointment với Doctor, PatientRegistration

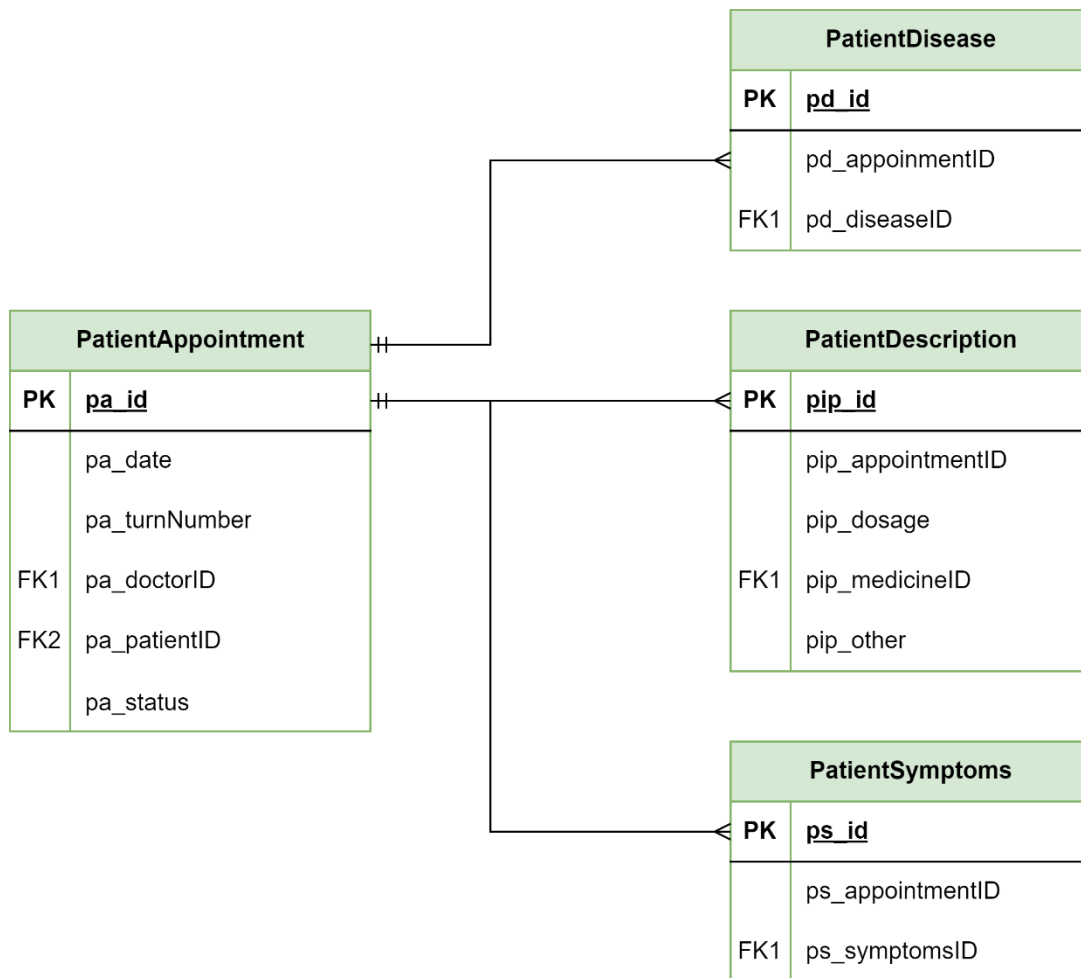
Mối quan hệ giữa PatientAppointment với Doctor:

- Một PatientAppointment chỉ chứa một Doctor
- Một Doctor có thể có ở nhiều PatientAppointment
- Biểu thị: PatientAppointment (1) ---- (N) Doctor

Mối quan hệ giữa PatientAppointment với PatientRegistration:

- Một PatientAppointment chỉ chứa một PatientRegistration
- Một PatientRegistration có thể có ở nhiều PatientAppointment
- Biểu thị: PatientAppointment (1) ---- (N) PatientRegistration

2.4.2.7 Mối quan hệ giữa thực thể PatientAppointment với PatientDisease



Hình 2.9 Mối quan hệ giữa PatientAppointment với PatientDisease

Mối quan hệ giữa PatientAppointment với PatientDisease:

- Một PatientDisease chỉ chứa một PatientAppointment
- Một PatientAppointment có thể có ở nhiều PatientDisease
- Biểu thị: PatientAppointment (N) ---- (1) PatientDisease

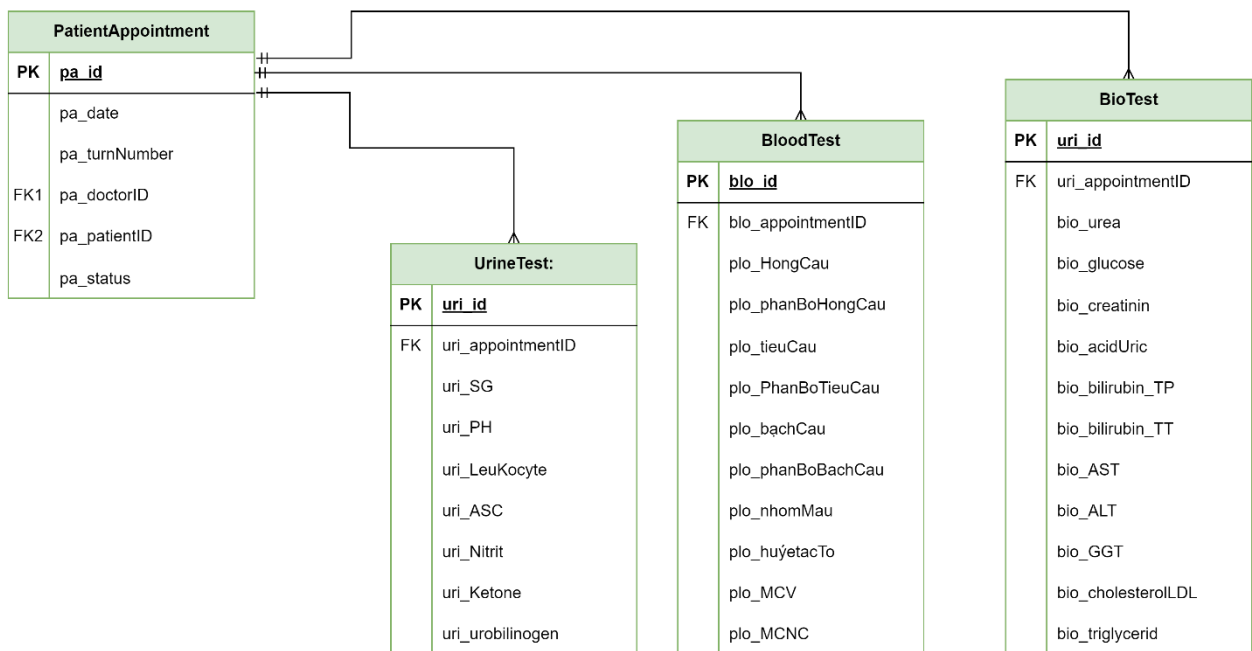
Mối quan hệ giữa PatientAppointment với PatientSymptoms:

- Một PatientSymptoms chỉ chứa một PatientAppointment
- Một PatientAppointment có thể có ở nhiều PatientSymptoms
- Biểu thị: PatientAppointment (N) ---- (1) PatientSymptom

Mối quan hệ giữa PatientAppointment với PatientDescription:

- Một PatientDescription chỉ chứa một PatientAppointment
- Một PatientAppointment có thể có ở nhiều PatientDescription
- Biểu thị: PatientAppointment (N) ---- (1) PatientDescription

2.4.2.8 Mối quan hệ giữa thực thể PatientAppointment với BioTest, UrineTest, BloodTest



Hình 2.10 Mối quan hệ giữa PatientAppointment với BioTest, BloodTest, UrineTest

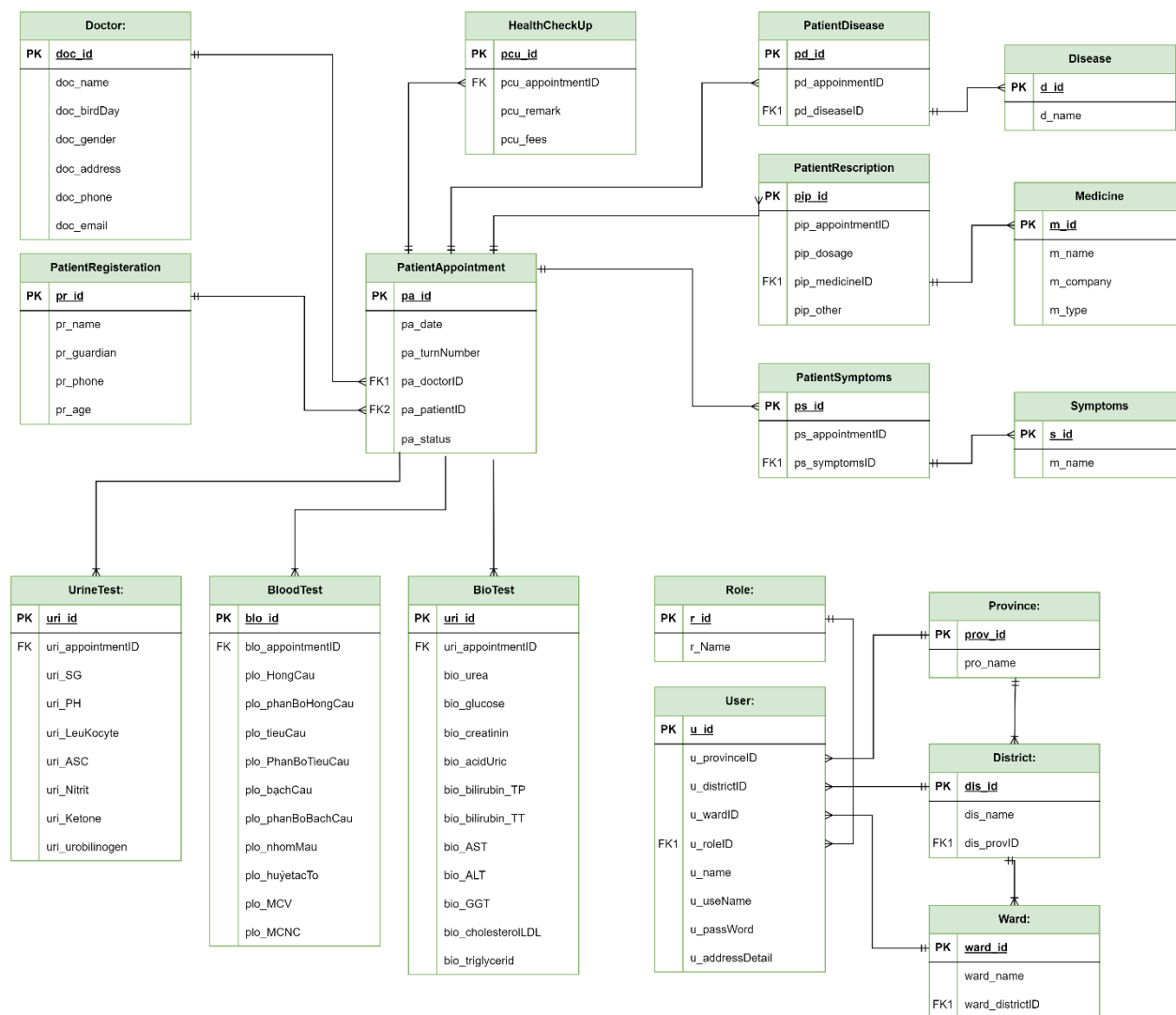
Mối quan hệ giữa PatientAppointment với UrineTest, BloodTest và BioTest là tương tự nhau:

- Một UrineTest chỉ chứa một PatientAppointment
- Một PatientAppointment có thể có ở nhiều UrineTest
- Biểu thị: PatientAppointment (N) ---- (1) PatientDisease

Mối quan hệ giữa PatientAppointment với BloodTest và BioTest là tương tự như với UrineTest.

2.4.3 Thiết kế ERD (Entity Relationship Diagram)

Sau khi bạn đã phân tích các thực thể cần cho hệ thống và các mối quan hệ giữa chúng, việc xây dựng biểu đồ ERD sẽ trở nên dễ dàng hơn nhiều. Biểu đồ ERD không chỉ giúp thể hiện cấu trúc cơ sở dữ liệu mà còn thể hiện được một cách trực quan về sự tương tác giữa các thực thể với nhau.



Hình 2.11 Thiết kế ERD

2.4.4 Cơ sở dữ liệu

2.4.4.1 Bảng Role (Vai trò)

Bảng 2.1 Cơ sở dữ liệu bảng Role(Vai trò)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	r_id	INT	Khoá chính	Không	Mã User
2	r_Name	CHAR		Không	Tên User

Đây là bảng chứa các role được sử dụng trong hệ thống.

2.4.4.2 Bảng User (Tài khoản người dùng)

Bảng 2.2 Cơ sở dữ liệu bảng User (Tài khoản người dùng)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	u_id	INT	Khoá chính	Không	Mã role
2	u_provinceID		Khóa ngoại	Không	Mã tỉnh thành
3	u_districtID		Khóa ngoại	Không	Mã quận
4	u_wardID		Khóa ngoại	Không	Mã phường
5	u_roleID		Khóa ngoại	Không	Mã vai trò
6	u_name			Không	tên user
7	u_useName			Không	Tên tài khoản
8	u_passWord			Không	Mật khẩu
9	u_addressDetail			Không	Chi tiết địa chỉ

Đây là bảng chứa thông tin của những cán bộ y tế được đăng kí sử dụng vào việc sử dụng các chức năng nghiệp vụ của mình

2.4.4.3 Bảng Doctor (Bác sĩ)

Bảng 2.3 Cơ sở dữ liệu bảng Doctor (Bác sĩ)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	doc_id	INT	Khoá chính	Không	Mã bác sĩ
2	doc_name	VARCHAR		Không	Tên bác sĩ
3	doc_birdDay	DATE		Không	Ngày sinh
4	doc_gender	INT		Không	Giới tính
5	doc_address	VARCHAR		Không	Địa chỉ
6	doc_phone	VARCHAR		Không	Số điện thoại
7	doc_email	VARCHAR		Không	Email

Đây là bảng chứa thông tin của các bác sĩ trong bệnh viện phục vụ cho việc đăng kí người khám cho bệnh nhân.

2.4.4.4 Bảng Nurse (Y tá)

Bảng 2.4 Cơ sở dữ liệu bảng Nurse (Y tá)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	nur_id	INT	Khoá chính	Không	Mã y tá
2	nur_name	CHAR		Không	Tên y tá
3	nur_birdDay	DATE		Không	Ngày sinh
4	nur_gender	INT		Không	Giới tính
5	nur_address	VARCHAR		Không	Địa chỉ
6	nur_phone	VARCHAR		Không	Số điện thoại
7	nur_email	VARCHAR		Không	Email

Đây là bảng chứa thông tin của các y tá trong bệnh viện phục vụ cho việc đăng kí người khám cho bệnh nhân.

2.4.4.5 PatientAppointment (Cuộc hẹn khám bệnh của bệnh nhân)

Bảng 2.5 Cơ sở dữ liệu bảng PatientAppointment (Cuộc hẹn khám bệnh của bệnh nhân)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	pa_id	INT	Khoá chính	Không	Mã cuộc hẹn khám bệnh
2	pa_date	FLOAT		Không	Ngày khám bệnh
3	pa_turnNumber	BIT		Không	Số thứ tự của cuộc hẹn
4	pa_doctorID	BIT	Khóa ngoại	Không	Mã bác sĩ
5	pa_patientID	BIT	Khóa ngoại	Không	Mã bệnh nhân
6	pa_status	BIT		Không	Trạng thái của buổi khám

Đây là bảng chứa thông tin các đăng kí của bệnh nhân trong mỗi lần đến thăm khám. Đây là một bảng quan trọng khi chứa rất nhiều thông tin dựa vào mã của PatientAppointment

2.4.4.6 Bảng Province (Tỉnh thành)

Bảng 2.6 Cơ sở dữ liệu bảng Province (Tỉnh thành)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	prov_id	INT	Khoá chính	Không	Mã tỉnh thành
2	pro_name	FLOAT		Không	Tên tỉnh thành

Đây là bảng chứa thông tin của 63 tỉnh thành được đưa trực tiếp vào database để hỗ trợ trong quá trình sử dụng phần mềm. Qua đó đem lại trải nghiệm tốt hơn cho người dùng. Dữ liệu cụ thể như:

- ID = 1, 'Ha Noi',
- ID = 2, 'Ho Chi Minh',

- ID = 3, 'Hai Phong'
- ID = 4, 'Da Nang'
- ID = 5, 'Can Tho'
-

2.4.4.7 Bảng Districst (Quận)

Bảng 2.7 Cơ sở dữ liệu bảng Districst (Quận)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	dis_id	INT	Khoá chính	Không	Mã quận
2	dis_name	FLOAT		Không	Tên quận
3	dis_provID	INT	Khóa ngoại	Không	Mã tỉnh thành

Đây là bảng chứa thông tin về quận/huyện và ứng với một tỉnh tương ứng tương tự như thông tin về tỉnh, thông tin của các quận huyện cũng được đưa trực tiếp vào database để hỗ trợ trong quá trình sử dụng phần mềm.

Dữ liệu cụ thể như:

- ID = 1, 'Huyện Ba Vì', ID_Convince = 1
- ID = 2, 'Huyện Chương Mỹ', ID_Convince = 1
- ID = 3, 'Huyện Đan Phượng', ID_Convince = 1
- ID = 4, 'Huyện Đông Anh', ID_Convince = 1
- ID = 5, 'Huyện Gia Lâm', ID_Convince = 1
-

2.4.4.8 Bảng Ward (Phường)

Bảng 2.8 Cơ sở dữ liệu bảng Ward (Phường)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	ward_id	INT	Khoá chính	Không	Mã phường
2	ward_name	FLOAT		Không	Tên phường
	ward_districtID	INT	Khóa ngoại	Không	Mã quận

Đây là bảng chứa thông tin về phường/xã và ứng với một quận/huyện tương ứng tương tự như thông tin về huyện, thông tin của các phường xã cũng được đưa trực tiếp vào database để hỗ trợ trong quá trình sử dụng phần mềm.

Dữ liệu cụ thể như:

- ID = 1, 'Xa Ba Trai', ID_District = 1
- ID = 2, 'Xa Cẩm Lĩnh', ID_District = 1
- ID = 3, 'Xa Chàng Sơn', ID_District = 1
-

2.4.4.9 Bảng PatientSymptoms (Triệu chứng của bệnh nhân)

Bảng 2.9 Cơ sở dữ liệu bảng PatientSymptoms (Triệu chứng của bệnh nhân)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	ps_id	INT	Khoá chính	Không	Mã
2	ps_appointmentID	FLOAT		Không	Mã buổi khám bệnh
	ps_symptomsID			Không	Mã triệu chứng bệnh

Đây là bảng chứa thông tin về bệnh mà bệnh nhân đang được điều trị hiện tại mắc phải. Mỗi bệnh nhân khám có thể có nhiều triệu chứng.

2.4.4.10 Bảng Medicine (Thuốc)

Bảng 2.10 Cơ sở dữ liệu bảng Medicine (Thuốc)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	m_id	INT	Khoá chính	Không	Mã thuốc
2	m_name	FLOAT		Không	Tên thuốc
3	m_company			Không	Tên công ty sản xuất
4	m_type			Không	Loại thuốc

Đây là bảng chứa thông tin về các loại thuốc thường được sử dụng sẽ được đưa vào database để cho việc lựa chọn dễ dàng và chính xác hơn

2.4.4.11 Bảng Disease (Bệnh)

Bảng 2.11 Cơ sở dữ liệu bảng Disease (Bệnh)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	d_id	INT	Khoá chính	Không	Mã bệnh
2	d_name	FLOAT		Không	Tên bệnh

Đây là bảng chứa thông tin về các loại bệnh thường gặp, sẽ được đưa vào database để cho việc lựa chọn dễ dàng và chính xác hơn

2.4.4.12 Bảng PatientRegistration (Bệnh nhân đăng ký)

Bảng 2.12 Cơ sở dữ liệu bảng PatientRegistration (Bệnh nhân đăng ký)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	pr_id	INT	Khoá chính	Không	
2	pr_name	FLOAT		Không	
3	pr_guardian	VARCHAR		Không	
4	pr_phone	VARCHAR		Không	
5	pr_age	INT		Không	
6	pr_cccd	VARCHAR		Không	

Đây là bảng chứa thông tin về bệnh nhân đến khám bệnh sẽ được xác định danh tính duy nhất bằng số điện thoại và căn cước công dân. Tức là mỗi CCCD là duy nhất.

2.4.4.13 Bảng PatientRescription (Đơn thuốc của bệnh nhân)

Bảng 2.13 Cơ sở dữ liệu bảng PatientRescription (Đơn thuốc của bệnh nhân)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	pip_id	INT	Khoá chính	Không	Mã đơn thuốc
2	pip_appointmentID	FLOAT		Không	Mã buổi hẹn khám bệnh
3	pip_dosage			Không	Liều lượng thuốc
4	pip_medicineID			Không	Mã thuốc

5	pip_other			Không	Khác
---	-----------	--	--	-------	------

Đây là bảng chứa thông tin về đơn thuốc và cách dùng sau được bác sĩ kê đơn tạo màn hình HealthCheckScreen.

2.4.4.14 Bảng PatientDisease (Kết quả khám bệnh của bệnh nhân)

Bảng 2.14 Cơ sở dữ liệu bảng PatientDisease (Kết quả khám bệnh của bệnh nhân)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	pd_id	INT	Khoá chính	Không	Mã kết quả kiểm tra
2	pd_appointmentID	FLOAT	Khóa ngoại	Không	Mã buổi hẹn khám bệnh
3	pd_diseaseID		Khóa ngoại	Không	Mã bệnh

Đây là bảng chứa thông tin về đơn thuốc và cách dùng sau được bác sĩ kê đơn tạo màn hình HealthCheckScreen.

2.4.4.15 Bảng UrineTest (Kết quả kiểm tra Urine)

Bảng 2.15 Cơ sở dữ liệu bảng UrineTest (Kết quả kiểm tra Urine)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỎNG	MÔ TẢ
1	uri_id	INT	Khoá chính	Không	Mã kết quả kiểm tra
2	uri_appointmentID	FLOAT		Không	Mã buổi hẹn khám bệnh
3	uri_SG			Không	
4	uri_PH			Không	
5	uri_LeuKocyte			Không	
6	uri_ASC			Không	
7	uri_Nitrit			Không	
8	uri_Ketone			Không	
9	uri_urobilinogen			Không	

Đây là bảng chứa thông tin về kết quả xét nghiệm Urine của bệnh nhân tại màn hình UrineTestScreen.

2.4.4.16 Bảng BloodTest (Kết quả kiểm tra máu)

Bảng 2.16 Cơ sở dữ liệu bảng BloodTest (Kết quả kiểm tra máu)

STT	TÊN	KIỂU DỮ LIỆU	RÀNG BUỘC	RỔNG	MÔ TẢ
1	blo_id	INT	Khoá chính	Không	Mã kết quả kiểm tra
2	blo_appointmentID	FLOAT		Không	Mã buổi hẹn khám bệnh
3	plo_HongCau			Không	
4	plo_tieuCau			Không	
5	plo_BachCau			Không	
6	plo_MCN			Không	
7	plo_MCV			Không	
8	plo_MCNC			Không	

Đây là bảng chứa thông tin về kết quả xét nghiệm Máu của bệnh nhân được ghi nhận tại màn hình BloodTestScreen.

2.5 Kết Luận

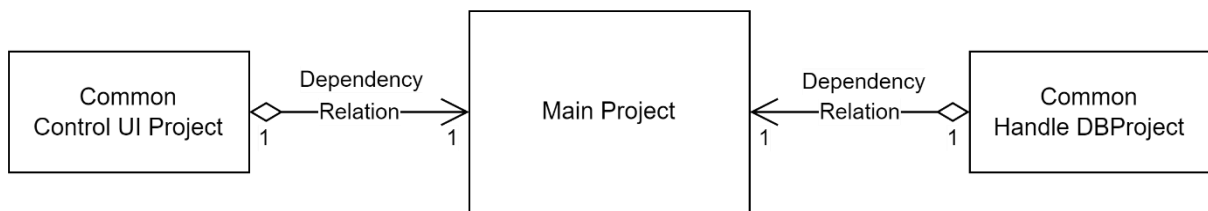
Như vậy sau các bước phân tích quy trình thực tế để xác định những thực thể tham gia vào trong hệ thống, những mối quan hệ tồn tại giữa chúng thì đã có được một cơ sở dữ liệu đáp ứng được cho nhu cầu mà hệ thống đặt ra. Tạo cơ sở cho việc xây dựng phần mềm, tổ chức phần mềm ở chương tiếp theo được rõ ràng và dễ dàng hơn. Cơ sở dữ liệu sẽ có thể được bổ sung trong tương lai theo yêu cầu thời điểm đó đặt ra.

CHƯƠNG 3. THIẾT KẾ VÀ TRIỂN KHAI PHẦN MỀM

Sau khi tìm hiểu các cơ sở lý thuyết ở chương 1, dựa trên các khảo sát và xác định yêu cầu chức năng phần mềm và xây dựng cơ sở dữ liệu chương 2, chương 3 sẽ tiến hành xây dựng phần mềm và kiểm thử.

3.1 Thiết kế phần mềm

3.2 Mô hình chương trình tổng quát



Hình 3.1 Mô hình chương trình tổng quát

Chia hệ thống thành 3 phần:

- **Main Winform Project:**
 - Đây là phần mềm trung tâm trong hệ thống, sẽ xây dựng giao diện để tương tác hỗ trợ trực tiếp cho người dùng.
 - Chịu trách nhiệm về việc hiển thị và nhận các yêu cầu từ phía người dùng
 - Sẽ được thêm các phương thức cung cấp từ 2 project chung là Common Control UI và Common Handle Project thông qua việc thêm vào các file thư viện .dll đã được đóng gói.
- **Common Control UI Project:**
 - Đây là project đại diện cho những việc xử lý các giao diện chung, như hiển thị màn hình chính và các màn hình thông báo, quản lý việc enable, disable, ...
 - Sẽ được biên dịch ra file: LibMainControl.dll
- **Common HandleD DB Project:**
 - Đây là project được gói các chức năng xử lý về logic để tiếp nhận và phản hồi lại cho chương trình chính (Main Winform).
 - Chứa các hàm giao tiếp trực tiếp với database để thêm sửa theo yêu cầu từ trên gọi xuống.

Chia nhỏ hệ thống thành các chương trình con với chức năng tách biệt được gói thành các API để thêm vào khi sử dụng đem lại nhiều ưu điểm và nhược điểm:

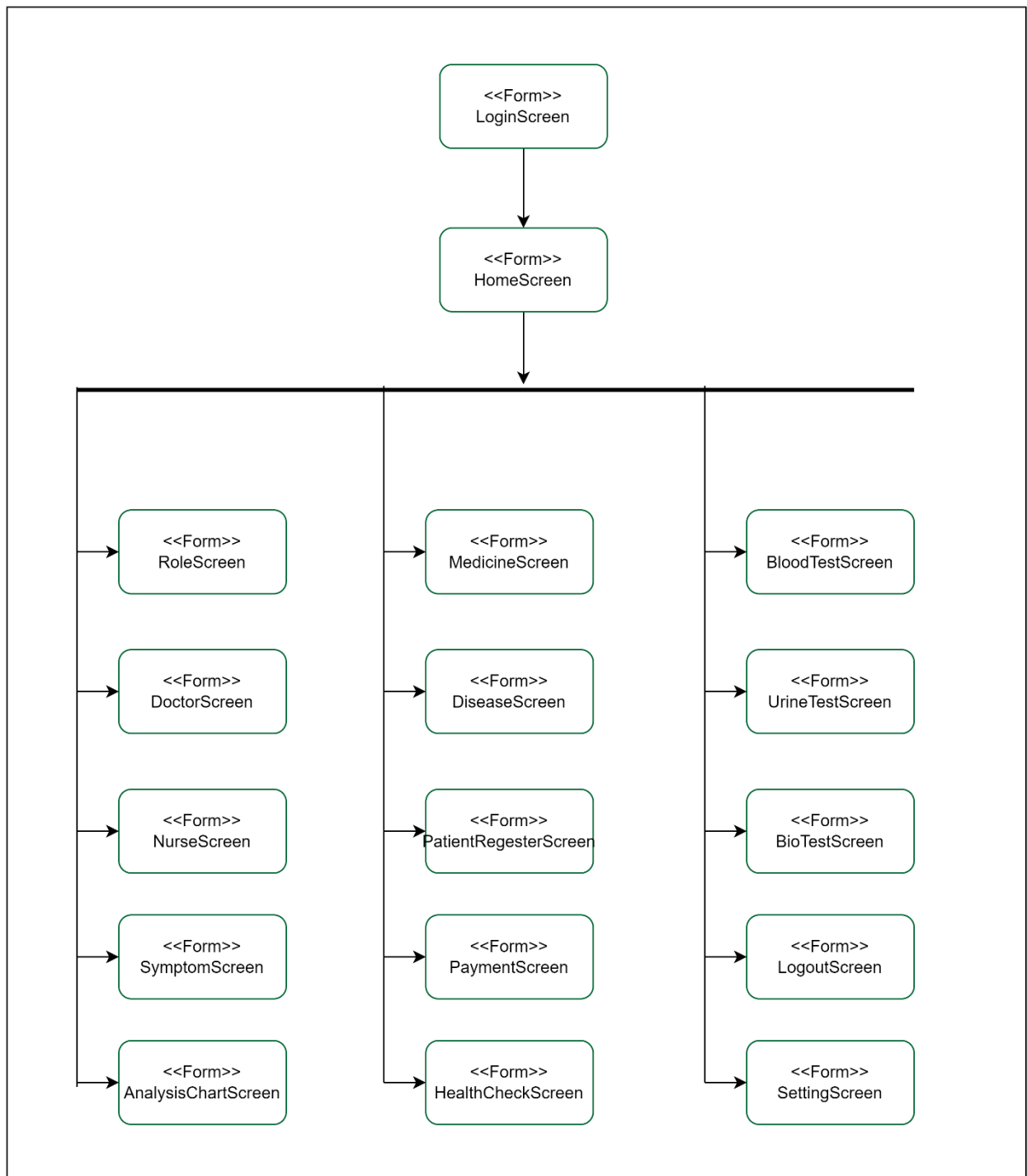
Về ưu điểm:

- Tổ chức quản lý source code: Vì mỗi project con có chức năng cụ thể nên về việc bảo trì và xác định vấn đề khi có lỗi là dễ dàng hơn khi hệ thống ngày một lớn hơn.
- Tái sử dụng code: các project con chứa các chức năng chung nên có thể sử dụng với nhiều dự án khác nhau.
- Khả năng mở rộng: có thể bổ sung chức năng mới và ứng dụng mà không bị ảnh hưởng đến các phần khác.
- Khả năng bảo trì hệ thống: Giảm thiểu tác động đến nhiều phần khi chỉ với một thay đổi nhỏ.

Về nhược điểm:

- Phức tạp trong việc triển khai và quản lý: Khi có nhiều project con chắc chắn sẽ khó về mặt quản lý chương trình.
- Giảm hiệu suất: Vì chia nhỏ chương trình nên việc phải tải các chương trình thứ 3 ngoài việc tăng kích thước thì hiệu năng có thể giảm.
- Khó khăn trong việc thay đổi kiến trúc: Khi thay đổi lớn có thể sẽ kéo theo sự thay đổi của các chương trình con rất khó để kiểm soát các vấn đề có thể gặp khi chạy chương trình.

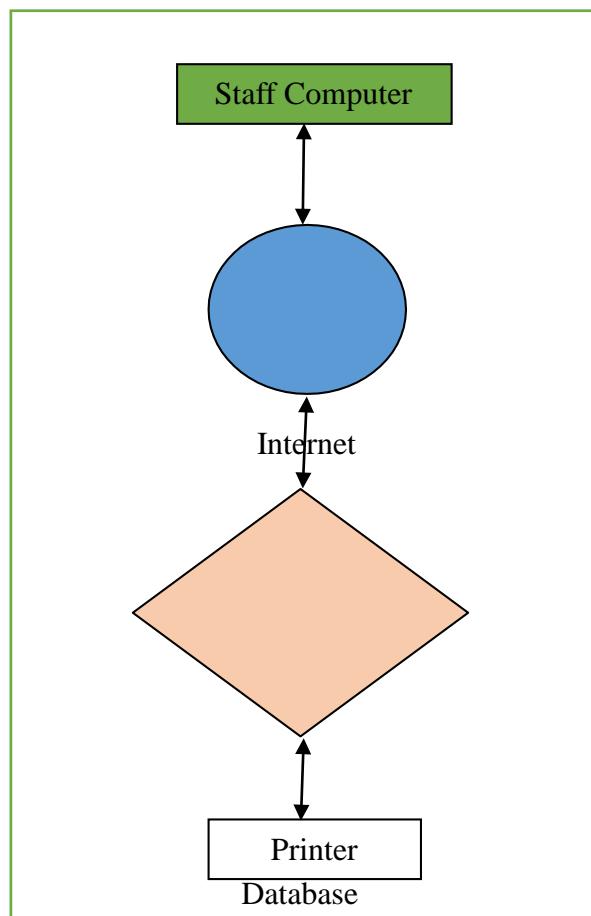
3.1.1 Window Navigation View Diagram



Hình 3.2 Window Negative Diagram

Đây là những màn hình cần thiết để có thể quản lý được tất cả thông tin của bệnh nhân sau một quy trình xét nghiệm và điều trị.

3.1.2 Net Work Model Diagram



Hình 3.3 Net Work Model Diagram

Chương trình sẽ cần chạy với kết nối mạng ổn định trong suốt quá trình chạy hệ thống mọi quy trình nhân viên y tế đều có thể thực hiện được từ khi bệnh nhân đăng kí khám bệnh đến khi in bệnh án hóa đơn làm thủ tục xuất viện

3.2 Xây dựng chương trình Main Winform Project

3.2.1 Thiết kế phần giao diện

Việc thiết kế giao diện đóng vai trò quan trọng trong việc xây dựng và phát triển ứng dụng ở thời điểm hiện tại. Đầu tiên vì đó là giao diện trực tiếp tương tác với người dùng nên cần phải thiết kế dựa trên nhu cầu và mong muốn của người dùng đảm bảo trải nghiệm người dùng tốt nhất. Sau đó là về việc tiện lợi cho việc lập trình phần mềm được hiệu quả nhất.

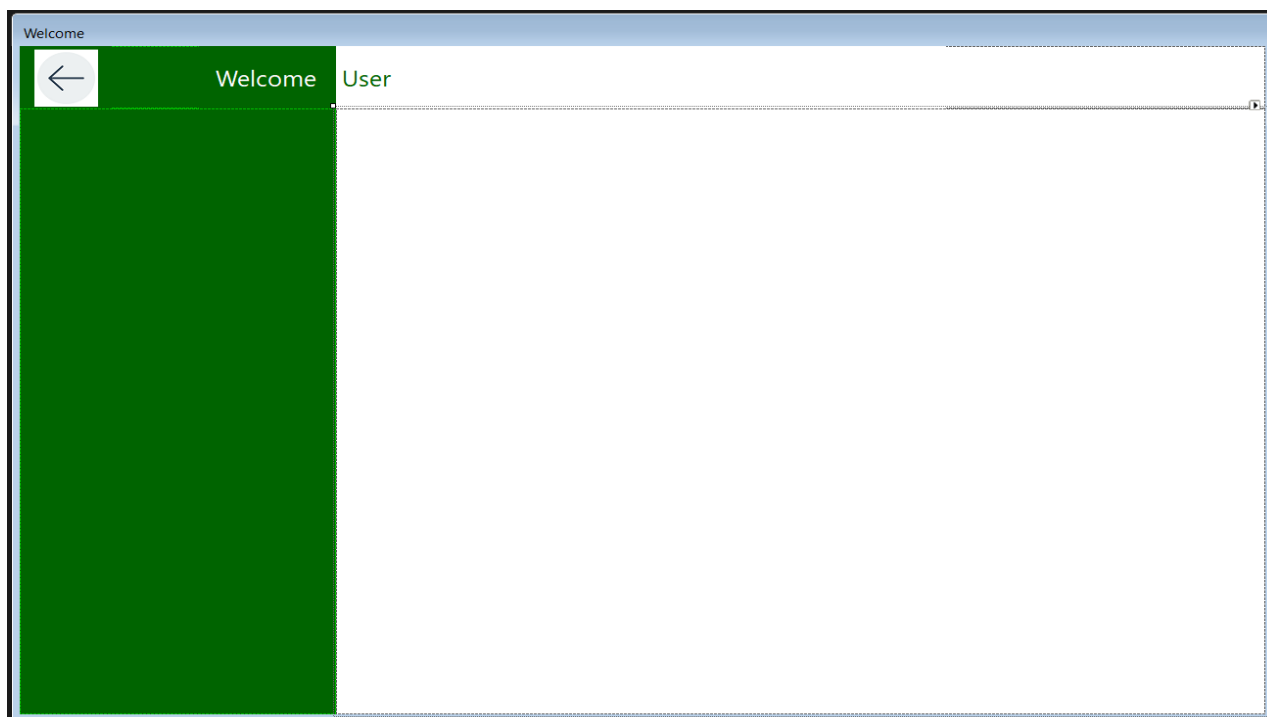
Vậy nên sử dụng kế thừa lúc này là vô cùng cần thiết với dự án hỗ trợ bệnh nhân điều trị thận nhân tạo. Khi áp dụng kế thừa một cách hợp lý, chúng ta có thể tái sử dụng các tính năng chung đã được xây dựng từ trước, giảm thiểu việc tạo mới những phần giống nhau. Một số điểm lợi thế của việc kế thừa.

1. Tái sử dụng mã nguồn
2. Tính nhất quán và đồng nhất
3. Dễ dàng mở rộng và sửa đổi

Sau khi phân tích cụ thể những yêu cầu cho phần mềm của mình. Em đưa ra các màn hình cơ sở như sau.

3.2.1.1 Các màn hình cơ sở

Đầu tiên là màn hình **Sample**:

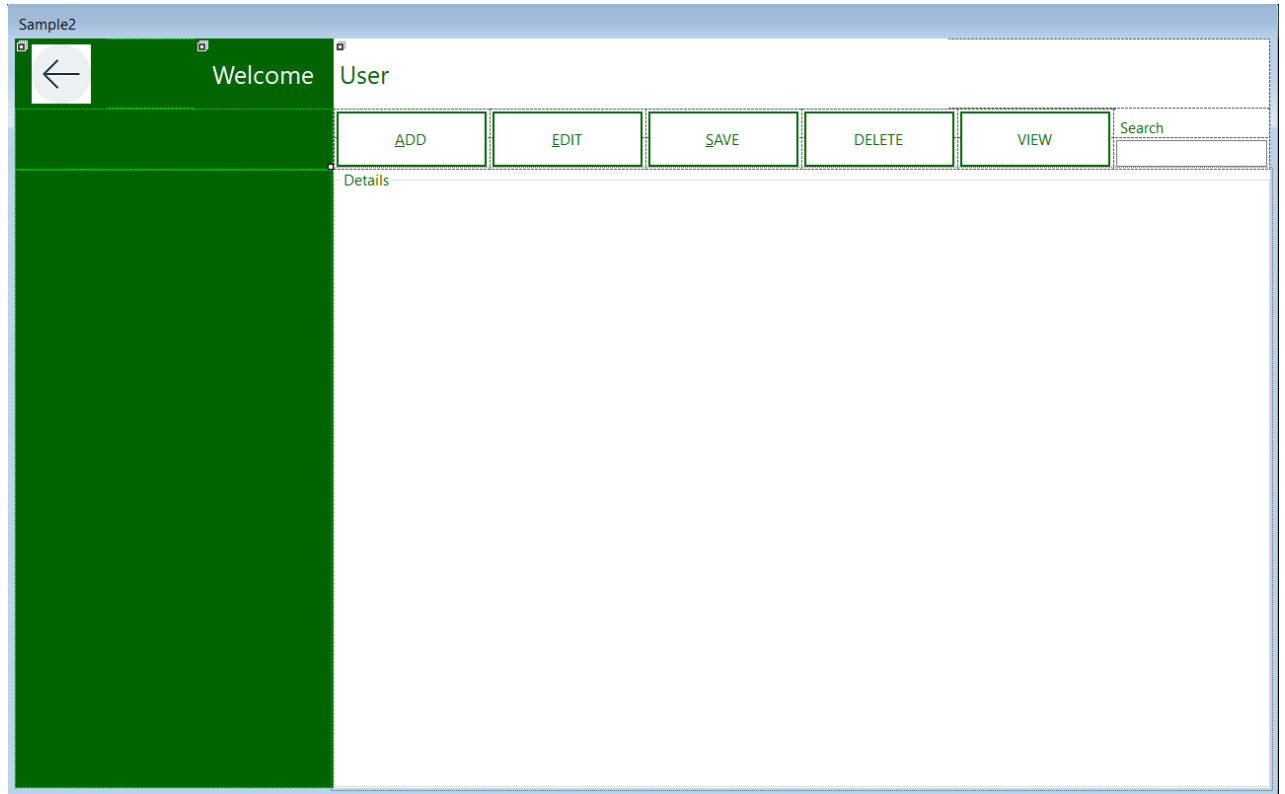


Hình 3.4 Màn hình Sample

Bảng 3.1 Các thành phần trên màn hình Sample

Type	Tên	Mô tả
Button	Back	Sẽ quay về trang Home
Label	User	Sẽ được Update theo tên của User đăng nhập vào App

Thứ hai là màn **Sample2** được kế thừa từ màn hình **Sample**:



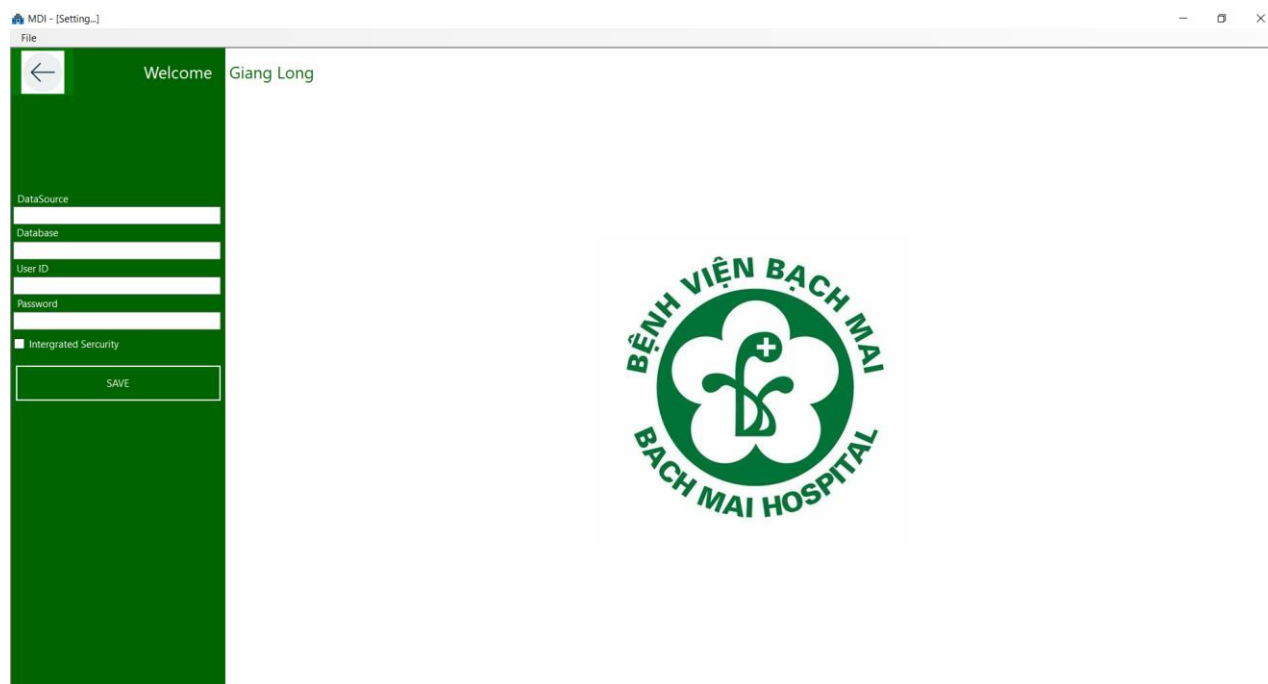
Hình 3.5 Sample2 Windows

Với việc bổ sung thêm các chức năng chung cho các màn hình có chức năng quản lý việc thêm sửa xóa trong Database.

Để đảm bảo cho tính nhất quán thì:

- Từ sau tất cả màn hình quản lý có chức năng thêm sửa xóa sẽ được kế thừa từ màn hình Sample2. Ví dụ như: RoleScreen, StaffScreen,...
- Và những màn hình không bao gồm chức năng thêm sửa xóa sẽ kế thừa màn hình Sample. Ví dụ như: LoginScreen, HomeScreen.

3.2.1.2 Màn hình SettingScreen



Hình 3.6 Màn hình SettingScreen

Màn hình Setting cho Admin là một giao diện quan trọng trong ứng dụng quản lý hệ thống, được thiết kế dành riêng cho của Admin. Giao diện này cho phép Admin cấu hình các thông tin liên quan đến kết nối tới cơ sở dữ liệu.

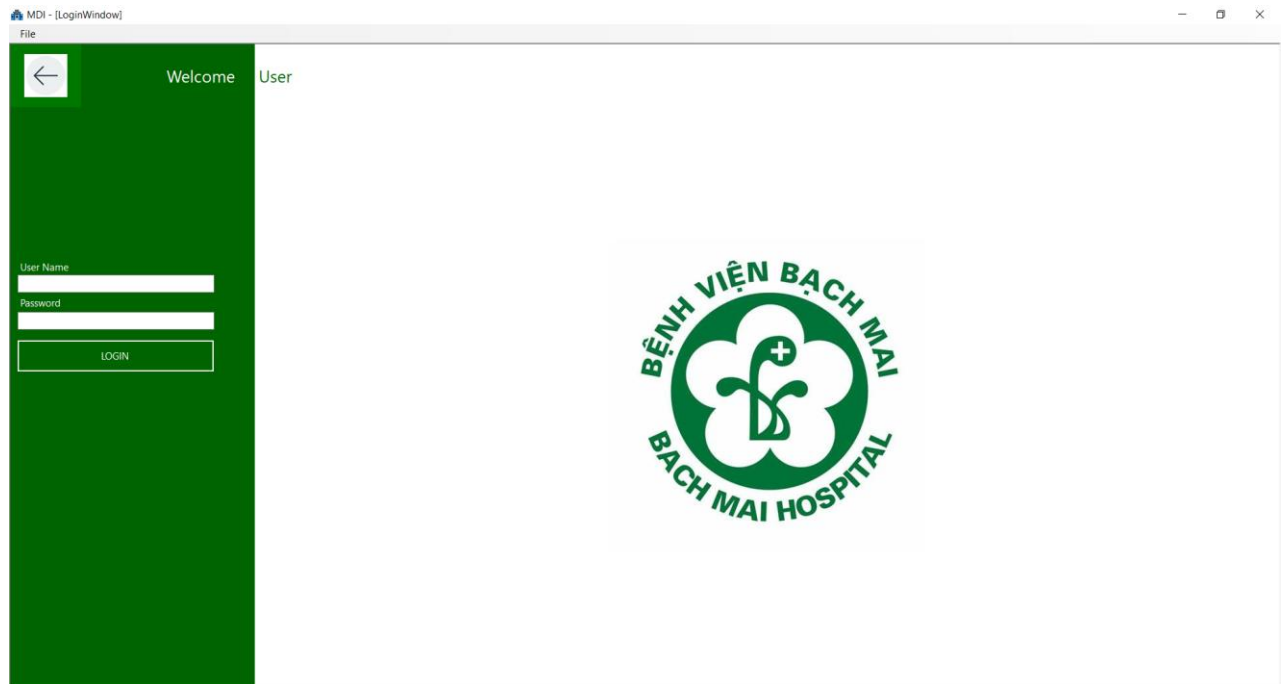
Yêu cầu cụ thể cho các thành phần:

Bảng 3.2 Các thành phần trên màn hình SettingScreen

Type	Tên	Mô tả
TextBox	DataSource	Nhập chuỗi IP và Port, cách nhau bởi dấu phẩy, để xác định vị trí của Server. Example: 172.19.200.89,1433
TextBox	Database	Nhập tên của CSDL muốn sử dụng ở trên Server.
TextBox	User ID	Nhập tài khoản có quyền mở kết nối vào máy chủ
TextBox	Password	Nhập mật khẩu cho tài khoản có quyền mở kết nối vào máy chủ
CheckBox	Intergrated Security	Cung cấp 2 cách đăng nhập vào CSDL cho Admin: 1) Khi checkbox được chọn: Trong trường hợp này thông tin tài khoản sẽ được lấy thông tin từ file mã hóa thông tin đã được đăng nhập lần trước đó. 2) Khi check box không được chọn: Trong trường hợp này Admin vẫn đăng nhập với tất cả thông tin. Và thông tin này sẽ được mã hóa

		ra file để dùng cho lần tiếp theo.
Button	Save	Lưu lại những thông tin cấu hình vừa được nhập phía trên Nếu có trường bỏ trống show MessageBox “Fields cannot Blank”.

3.2.1.3 Màn hình LoginScreen



Hình 3.7 Màn hình LoginScreen

Giao diện đăng nhập này giúp người dùng xác thực bằng 2 trường Username và Password để truy cập vào hệ thống. Với mỗi tài khoản khác nhau sẽ sử dụng được chức năng tương ứng với Role của họ.

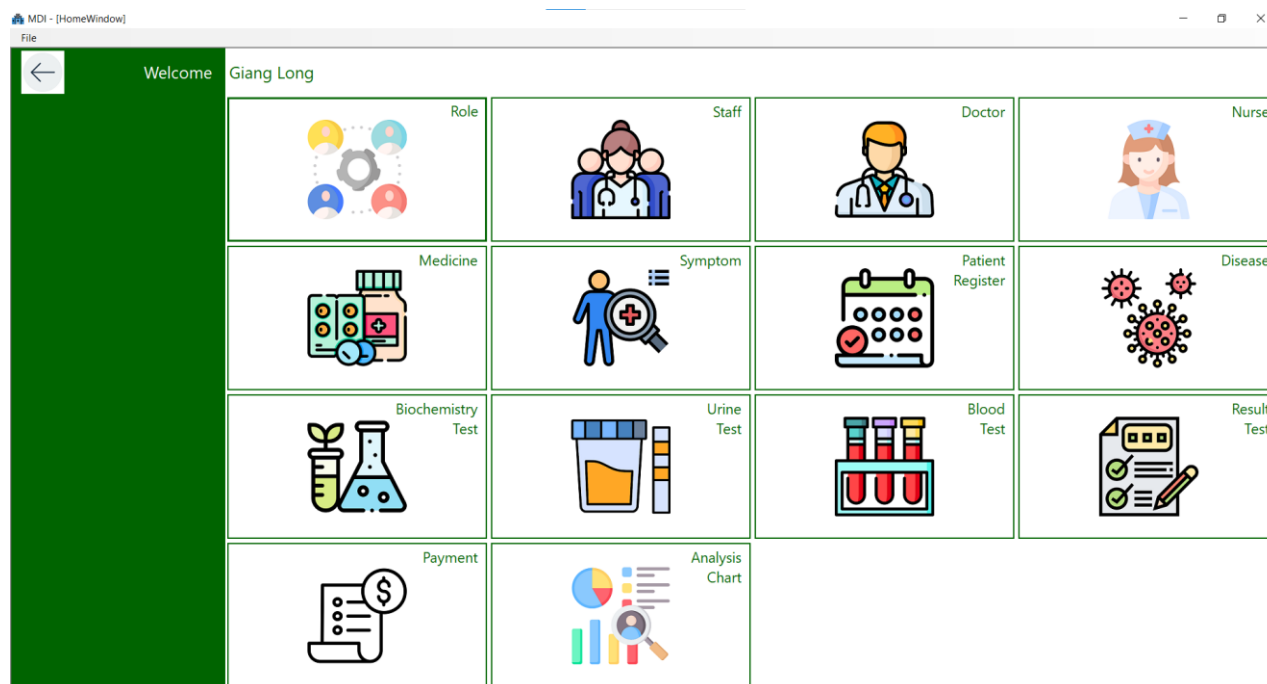
Yêu cầu từng thành phần:

Bảng 3.3 Các thành phần trên màn hình LoginScreen

Type	Tên	Mô tả
TextBox	User Name	Nhập tên tài khoản được cấp của người dùng.
TextBox	Password	Nhập mật khẩu cho tài khoản được cấp của người dùng.
Button	Login	Xác thực thông tin tài khoản vừa được nhập có tồn tại trong hệ thống thì chuyển vào màn hình chính. 1) Nếu có trường để trống, show MessageBox: “Fields cannot blank”.

		2) Nếu thông tin tài khoản không hợp lệ thì cho phép nhập lại.
--	--	--

3.2.1.4 Màn hình HomeScreen

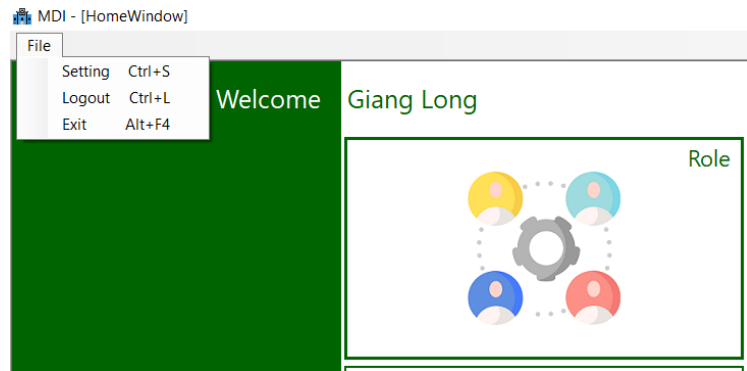


Hình 3.8 Màn hình HomeScreen

Sau khi đăng nhập với thông tin đăng nhập đúng tại màn hình login người dùng sẽ được chuyển sang màn hình HomeScreen. Đây là màn hình chứa tất cả các chức năng mà phần mềm có thể sử dụng.

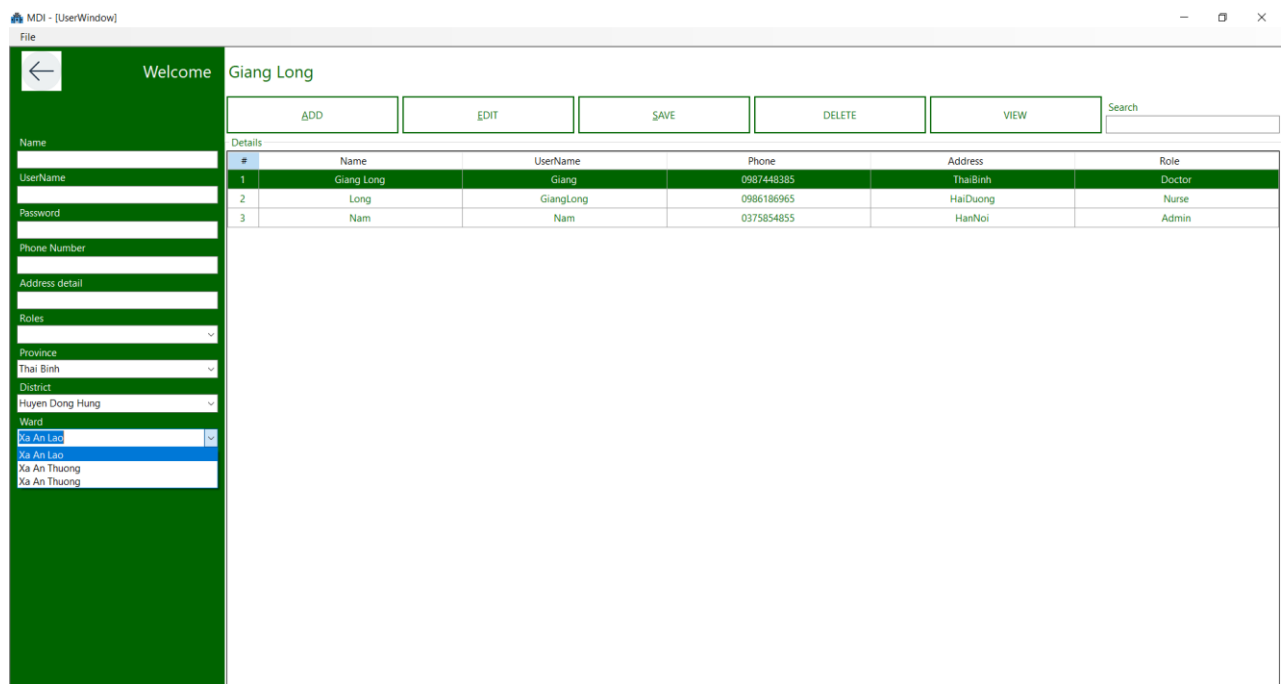
Yêu cầu về các thành phần có trên màn hình:

- Có 16 nút bấm để gọi đến các màn hình tương ứng là :RoleScreen, StaffScreen, DoctorScreen, NurseScreen, MedicineScreen, SymptomScreen, PatientRegistrationScreen, DiseaseScreen, BioTestScreen, UrineTestScreen, BloodTestScreen, ResultTestScreen, PaymentScreen, AnalysisChartScreen.
- Sẽ có thêm các chức năng mở rộng khi bấm nút File trên thanh Control bao gồm : SettingScreen, Logout action, Exit action.



Hình 3.9 Màn hình con File button

3.2.1.5 Màn hình StaffScreen

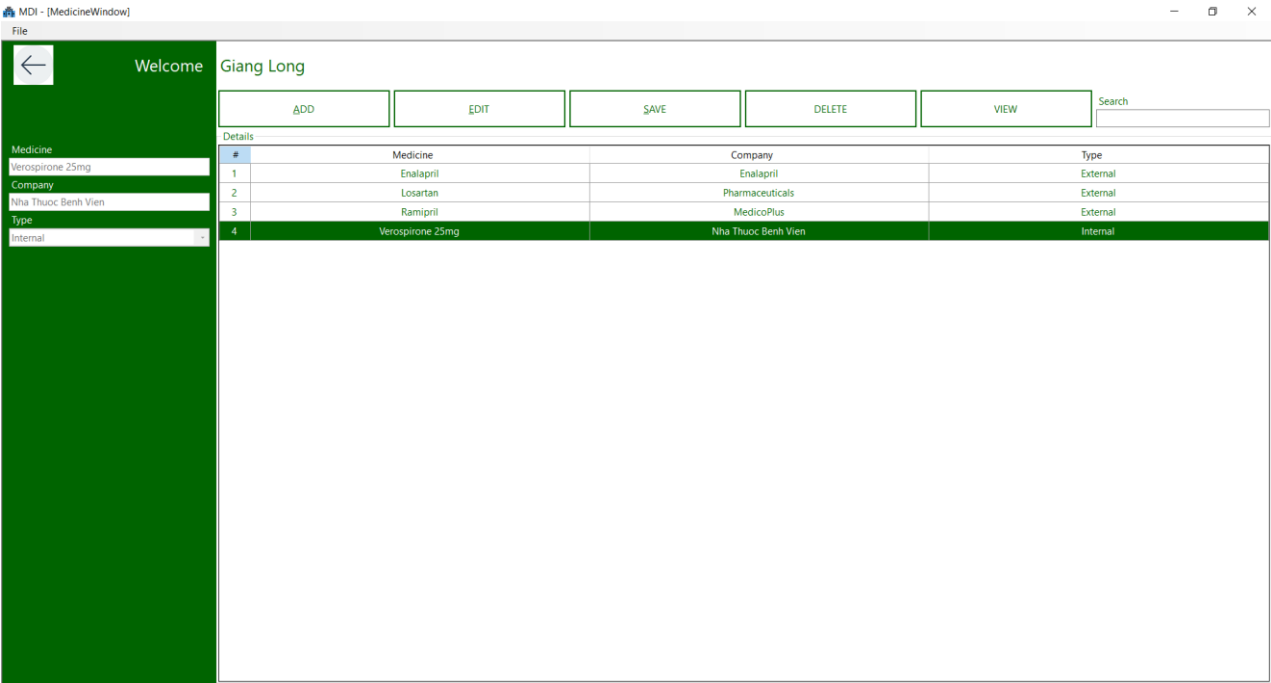


Hình 3.9 Màn hình StaffScreen

Đây là màn hình dành cho admin đăng kí tài khoản sử dụng cho các bác sĩ, y tá làm việc trực tiếp với các phòng ban xét nghiệm chuẩn đoán cho bệnh nhân điều trị thận nhân tạo.

Hành vi của các đối tượng trong màn hình:

3.2.1.6 Màn hình MedicineScreen



Hình 3.10 Màn hình MedicineScreen

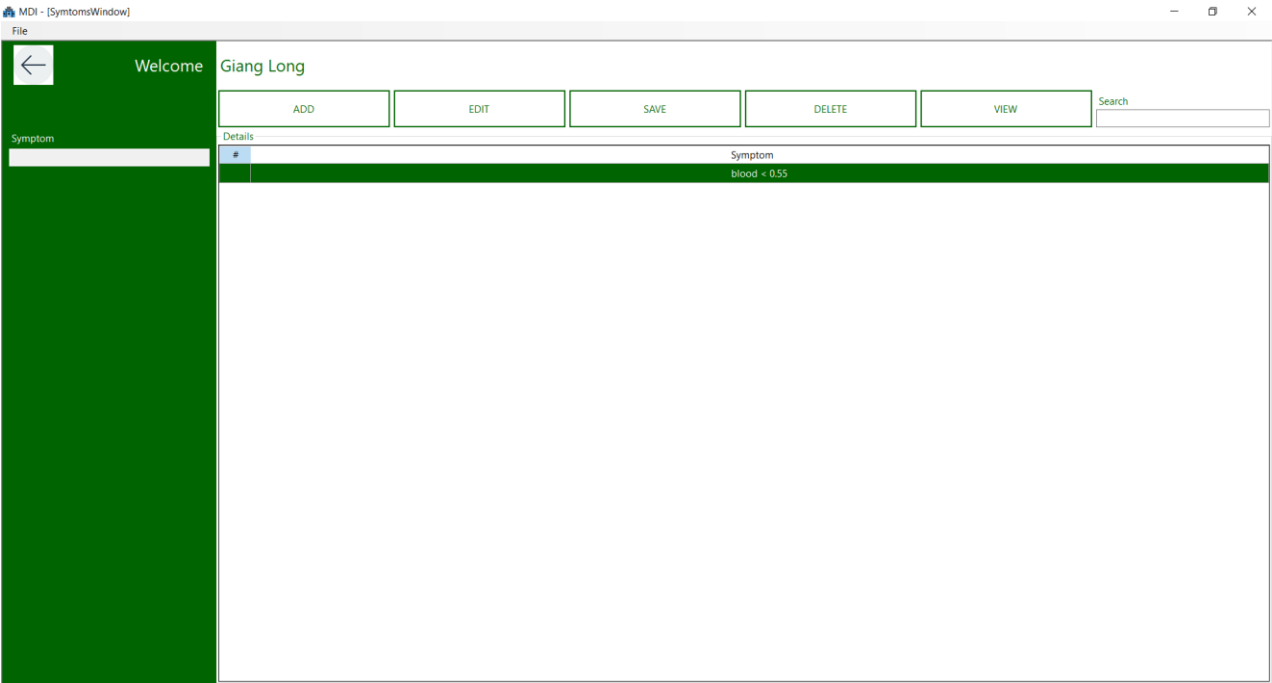
Màn hình MedicinScreen sẽ có các hoạt động thêm sửa xóa thông tin về các loại thuốc dùng để điều trị cho bệnh nhân, bao gồm các thông tin như: tên thuốc, công ty cung cấp và loại thuốc ngoại hay nội.

Yêu cầu về từng thành phần:

Bảng 3.4 Các thành phần trên màn hình MedicineScreen

Type	Tên	Mô tả
TextBox	Medicine	Nhập tên loại thuốc
TextBox	Company	Nhập đơn vị cung cấp loại thuốc này
ComboBox	Type	Lựa chọn loại thuốc: 1) Internal: Thuốc sản xuất trong nước 2) Thuốc nhập khẩu

3.2.1.7 Màn hình SysptomScreen



Hình 3.11 Màn hình SysptomScreen

Màn hình SymptomScreen được sử dụng để quản lý thông tin về các triệu chứng trong việc đưa ra những kết luận cho bệnh nhân điều trị thân nhân tạo. Giao diện này cho phép thêm thông tin về một triệu chứng thường gặp và lựa chọn bệnh tương ứng thông qua ComboBox Disease.

Yêu cầu về từng thành phần:

Bảng 3.5 Các thành phần trên màn hình SysptomScreen

Type	Tên	Mô tả
TextBox	Symptom	Nhập triệu chứng thường gặp
ComboBox	Disease	Lựa chọn bệnh mà bệnh nhân đang gặp phải

3.2.1.8 Màn hình BioTestScreen

←

Welcome

Date

14-Jul-2023

Select Patient

Andrew

Patient History

Phone

0375854855

Age

23

Bio Test Result

Urea

cm

Glucose

cm

Creatinin

cm

Acid Uric

cm

Bilirubin TP

cm

Bilirubin TT

cm

AST

cm

ALT

cm

GGT

cm

Cholesterol HDL

cm

Cholesterol LDL

cm

Triglycerid

cm

User

ADD

EDIT

SAVE

DELETE

VIEW

Search

Details

#	Appointment For	Urea	Glucose	Creatinin	Acid Uric	Bilirubin TP	Bilirubin TT	AST	ALT	GGT	Choles HDL	Choles LDL	Triglycerid
1	14-Jul-2023	0.1	0.99	0.2	0.1	0.2	0.2	0.1	0.2	0.2	0.1	0.2	0.2

Hình 3.12 Màn hình BioTestScreen

Màn hình BioTestScreen sẽ được các Y tá, Bác sĩ tại phòng xét nghiệm hóa sinh sử dụng để thực hiện chức năng thêm, sửa và xóa với các thông tin xét nghiệm hóa sinh của bệnh nhân.

Với màn hình lưu thông tin xét nghiệm việc thêm sửa xóa là có sự khác biệt với những màn hình khác, cơ bản sẽ đi qua các bước sau để thêm thông tin cho một bệnh nhân:

- **Step1:** Chọn ngày tại DateTime
Thông tin các lịch đăng kí của bệnh chưa được hoàn thành việc thăm khám sẽ hiện ra ở ComboBox SelectPatient.
- **Step2:** Chọn một bệnh nhân trong selectPatient
Thông tin về lịch sử thăm khám của bệnh nhân sẽ hiện ra (Những trường này sẽ bị disable).
- **Step3:** Nhập thông tin xét nghiệm từ máy xét nghiệm
- **Step4:** Nhấn Save Button

Bảng 3.6 Các thành phần trên màn hình BioTestScreen

Type	Tên	Mô tả
DateTimePicker	Date	Nhập ngày làm để làm xét nghiệm hóa sinh

		cho các bệnh nhân.
ComboBox	Select Patient	Hiện thị danh sách bệnh nhân đăng kí làm xét nghiệm hóa sinh vào ngày được chọn ở TextBox Date
Button	Call Patient	Bấm để tải danh sách các bệnh nhân cho nút Select Patient.
TextBox	Phone	Hiện thị số điện thoại đăng kí của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Age	Hiện thị tuổi của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Urea	Check kiểu dữ liệu input là float. Nếu không phải show Warning Message “Input type not correct”
TextBox	Glucose	Như yêu cầu bên trên
TextBox	Acid Uric	Như yêu cầu bên trên
TextBox	Creatinin	Như yêu cầu bên trên
TextBox	Bilirubin_TP	Như yêu cầu bên trên
TextBox	Bilirubin_TT	Như yêu cầu bên trên
TextBox	AST	Như yêu cầu bên trên
TextBox	ALT	Như yêu cầu bên trên
TextBox	GGT	Như yêu cầu bên trên
TextBox	Cholesterol_HDL	Như yêu cầu bên trên
TextBox	Cholesterol_LDL	Như yêu cầu bên trên
TextBox	Triglycerid	Như yêu cầu bên trên

3.2.1.9 Màn hình UrineTestScreen

←

Welcome

Date

14-Jul-2023

Select Patient

Andrew

Patient History

Phone

0375854855

Age

23

Urine Test Result

Specific Gravity	PH	ASC-Sỏi Cặn
<div>cm</div>	<div>cm</div>	<div>cm</div>
Leukocytes	Nitrit	Ketone
<div>● Âm tính</div> <div>● Dương Tính</div>	<div>● Âm tính</div> <div>● Dương Tính</div>	<div>● <0.5 mmol/L</div> <div>● >=0.5 mmol/L</div>
Urobilinogen		
<div>● Không Có</div> <div>● Có</div>		

User

ADD

EDIT

SAVE

DELETE

VIEW

Search

#	Appointment For	Specific Gravity	PH	ASC	Leukocytes	Nitrit	Ketone	Urobilinogen
1	14-Jul-2023 08:02 PM	0.21	0.55	0.3	Âm Tính	Âm Tính	< 0.5 mmol/L	Không Có

Hình 3.13 Màn hình UrineTestScreen

Màn hình UrineTestScreen được dùng cho các Y tá, Bác sĩ tại phòng xét nghiệm nước tiểu, dùng để thực hiện các hoạt động thêm sửa xóa các thông số xét nghiệm nước tiểu của bệnh nhân điều trị thận.

Bảng 3.7 Các thành phần trên màn hình UrineTestScreen

Type	Tên	Mô tả
DateTimePicker	Date	Nhập ngày làm để làm xét nghiệm nước tiểu cho các bệnh nhân.
ComboBox	Select Patient	Hiện thi danh sách bệnh nhân đăng kí làm xét nghiệm nước tiểu vào ngày được chọn ở TextBox Date
Button	Call Patient	Bấm để tải danh sách các bệnh nhân cần làm xét nghiệm vào cho Combobox Select Patient.
TextBox	Phone	Hiện thị số điện thoại đăng kí của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Age	Hiện thị tuổi của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Specific Gravity	Check kiểu dữ liệu input là float. Nếu không phải show Warning Message “Input type not correct”

TextBox	PH	Như trên
TextBox	ASC – Sỏi cặn	Như trên
2 Radio Button	Leukocytes	Dùng để lựa chọn kết quả cho thông số xét nghiệm bao gồm 2 lựa chọn: Âm tính Dương tính
2 Radio Button	Nitrit	Như trên
2 Radio Button	Ketone	Dùng để lựa chọn kết quả cho thông số xét nghiệm bao gồm 2 lựa chọn: < 0.05 mmol/L >= 0.05 mmol/L
2 Radio Button	urobilinogen	Dùng để lựa chọn kết quả cho thông số xét nghiệm bao gồm 2 lựa chọn: Không có Dương tính

Với màn hình lưu thông tin xét nghiệm Urie, thì cũng tương tự như màn hình BioTest:

- **Step1:** Chọn ngày tại DateTime
Thông tin các lịch đăng kí của bệnh chưa được hoàn thành việc thăm khám sẽ hiện ra ở ComboBox SelectPatient.
- **Step2:** Chọn một bệnh nhân trong selectPatient
Thông tin về lịch sử thăm khám của bệnh nhân sẽ hiện ra (Những trường này sẽ bị disable).
- **Step3:** Nhập thông tin xét nghiệm từ máy xét nghiệm và tích vào trường phù hợp với thông tin từ máy
- **Step4:** Nhấn Save Button

3.2.1.10 Màn hình BloodTestScreen

The screenshot shows the BloodTestScreen application. The left sidebar is green and contains the following elements:

- Welcome** section with a **User** dropdown.
- Date** picker set to 14-Jul-2023.
- Select Patient** dropdown.
- CALL PATIENT** button.
- Patient History** section with **Phone** and **Age** input fields.
- Blood Test Result** section with a table of test results.

The main content area displays a **Details** table with the following data:

#	Appointment For	Hồng Cầu	Phân Bô Hồng Cầu	Bạch Cầu	Phân Bô Bạch Cầu	Tiểu Cầu	Phân Bô Tiểu Cầu	Nhóm Máu	Huyết Sắc Tố	MCV	MCN	MCNC
1	14-Jul-2023 08:02 PM	0.1	0.2	0.1	0.12	0.1	0.12	0.1	0.12	0.13	0.14	0.13

Hình 3.14 Màn hình BloodTestScreen

Màn hình BloodTestScreen được dùng cho các Y tá, Bác sĩ tại phòng xét máu, dùng để thực hiện các hoạt động thêm sửa xóa các thông số xét nghiệm máu của bệnh nhân điều trị thận.

Bảng 3.8 Các thành phần trên màn hình BloodTestScreen

Type	Tên	Mô tả
DateTimePicker	Date	Nhập ngày làm để làm xét nghiệm nước tiểu cho các bệnh nhân.
ComboBox	Select Patient	Hiện thi danh sách bệnh nhân đăng kí làm xét nghiệm nước tiểu vào ngày được chọn ở TextBox Date
Button	Call Patient	Bấm để tải danh sách các bệnh nhân cần làm xét nghiệm vào cho Combobox Select Patient.
TextBox	Phone	Hiện thị số điện thoại đăng kí của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Age	Hiện thị tuổi của bệnh nhân Luôn luôn bị Disable ở màn hình này.
TextBox	Hồng Cầu	Check kiểu dữ liệu input là float. Nếu không phải show Warning Message “Input type not correct”

TextBox	Độ phân bố hồng cầu	Như yêu cầu bên trên
TextBox	Tiểu cầu	Như yêu cầu bên trên
TextBox	Độ phân bố tiểu cầu	Như yêu cầu bên trên
TextBox	Bạch cầu	Như yêu cầu bên trên
TextBox	Nhóm máu	Như yêu cầu bên trên
TextBox	Huyết sắc tố	Như yêu cầu bên trên
TextBox	MCV	Như yêu cầu bên trên
TextBox	MCN	Như yêu cầu bên trên
TextBox	MCNC	Như yêu cầu bên trên

Với các chức năng quản lý thông tin của Blood test là hoàn toàn giống với các hành vi ở màn hình Bio test.

3.2.1.11 Màn hình HealthCheckScreen

Hình 3.15 Màn hình HealthScreen

Đây là màn hình dành cho bác sĩ khi bệnh nhân đã có kết quả cả 3 xét nghiệm bác sĩ có thể lấy thông tin các triệu chứng từ bệnh nhân, thông tin kết quả xét nghiệm, từ đó đưa ra kết luận về tình trạng bệnh. Bên cạnh đó kê thuốc điều trị phù hợp cho bệnh nhân kèm tên thuốc và cách dùng.

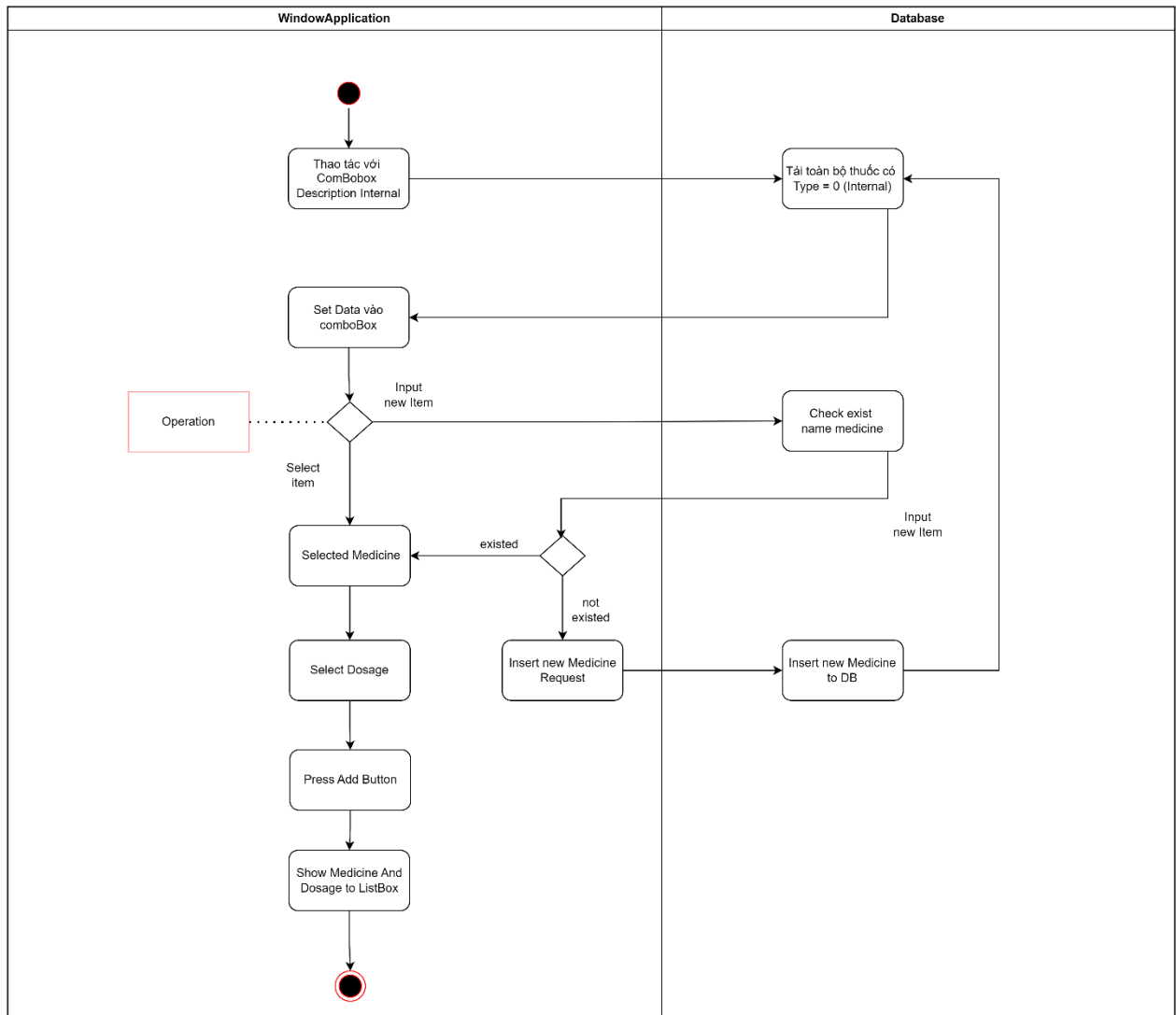
Tại màn hình này sau khi bác sĩ đã chọn các kết luận và lưu lại thì lượt khám này sẽ được chuyển sang trạng thái đã đánh giá để chuyển sang bước lấy hóa đơn và thanh toán viện phí tại màn hình PaymentScreen.

Tại đây bác sĩ sẽ có 6 trường cần thêm thông tin trước khi lưu thông tin xuống database và chuyển sang bước tiếp theo:

- **PatientDisease (1):** Thêm triệu chứng của bệnh nhân đang gặp
- **PatientSymptom (2):** Thêm chuẩn đoán về bệnh mà bệnh nhân đang gặp
- **PatientDescriptionInternal (3):** Thêm đơn thuốc nội cho bệnh nhân
- **PatientDescriptionExternal (4):** Thêm đơn thuốc ngoại cho bệnh nhân
- **Remark (5):** Thêm phần chú thích về bệnh tình cho bệnh nhân
- **Fees (6):** Thêm số tiền phí mà bệnh nhân cần chi trả

Hành động để thêm thông tin cho các trường (1), (2), (3), (4) là tương tự như nhau, dưới đây là mô tả cho hành động với **PatientDescriptionInternal**:

- Thêm đơn thuốc:



Hình 3.17 Activity for Add MedicineInternal

Tương tự như **MedicineInternal (3)**, sau khi Add đầy đủ thông tin ở 6 trường thông tin được liệt kê phía trên thì có thể bấm nút ADD control thành công cho màn hình HealthCheckupScreen để hoàn thành bước chuẩn đoán kê đơn thuốc cho bệnh nhân để bắt đầu chuyển sang bước in bệnh án và xuất viện.

3.2.1.12 Màn Hình PaymentScreen

#	Patient	Guardian	Age	Phone	Date Time	Doctor	Status	Action
1	Andrew	AndrewFather	23	0375854855	14-Jul-2023 12:00 AM	Giang Long	Consulted	Print

Hình 3.16 Màn hình PaymentScreen

Tại màn hình này bác sĩ, y tá phụ trách có thể thực hiện tổng hợp thông tin về bệnh và thuốc đã được kê kèm giá thành từng loại cho bệnh nhân tiến hành nhận thuốc và thanh toán hóa đơn cho bệnh nhân.

Với màn hình này hành động thực hiện như sau:

- **Step1:** Chọn ngày tại DateTime
- **Step2:** Nhấn Load button : Thông tin các bệnh nhân ở trạng thái đã được chuẩn đoán tại màn hình HealthyCheckup phía trên.
- **Step3:** Nhấn vào Action – Print của bệnh nhân cần in hóa đơn và bệnh án , sau đó màn hình PrintControler hiển thị. Bấm Print để in trong trường hợp đã kết nối với máy in.

Như vậy sau màn này là hoàn tất thủ tục khám bệnh cho một lượt thăm khám với người bệnh.

3.2.2 Xây dựng lớp Log Contronller

3.2.2.1 Khái niệm chung

Log là các bản ghi, dữ liệu về sự kiện xảy ra trong khi ứng dụng hoạt động. Đây là một phần quan trọng ở trong các ứng dụng khi chương trình ngày càng được mở rộng với chức năng phức tạp.

Trong quá trình hoạt động luôn luôn sẽ có những sự cố xảy ra, các ngoại lệ chưa tính đến và khi đó log thể hiện vai trò của mình. Nó giúp tìm ra nguyên nhân sự cố giúp bạn khắc phục nó. Trong môi trường develop hoàn toàn có thể sử dụng các công cụ mà IDE hỗ trợ để debug. Nhưng trên môi trường product không thể tái hiện được lỗi xảy ra. Vì vậy lúc này ta sử dụng log để xem lại lịch sử làm việc của phần mềm gặp sự cố ở đâu.

3.2.2.2 Xây dựng chương trình

Hiện tại project chỉ dùng một chế độ log duy nhất để ghi lại quá trình thêm sửa xóa thông tin với database.

Trong Class LogControler:

- **static public void WriteLog(string log):** Đây là hàm phụ trách việc ghi log ra file khi chế độ ghi log được bật
- Tên file: “hospital_ymmdd.txt”, trong đó **ymmdd** là năm tháng ngày. Ví dụ: hospital_230724.txt.
- Fomat cho dữ liệu trong Log: [Current Timestamp], [Log Message]. Ví dụ dữ liệu như dưới đây:

```
===== Hospital System =====
```

```
Version: 1.0.0
```

```
=====
```

```
2023-07-24 15:25:45.123, Application started.
```

```
2023-07-24 15:30:10.987, User 'Giang' logged in.
```

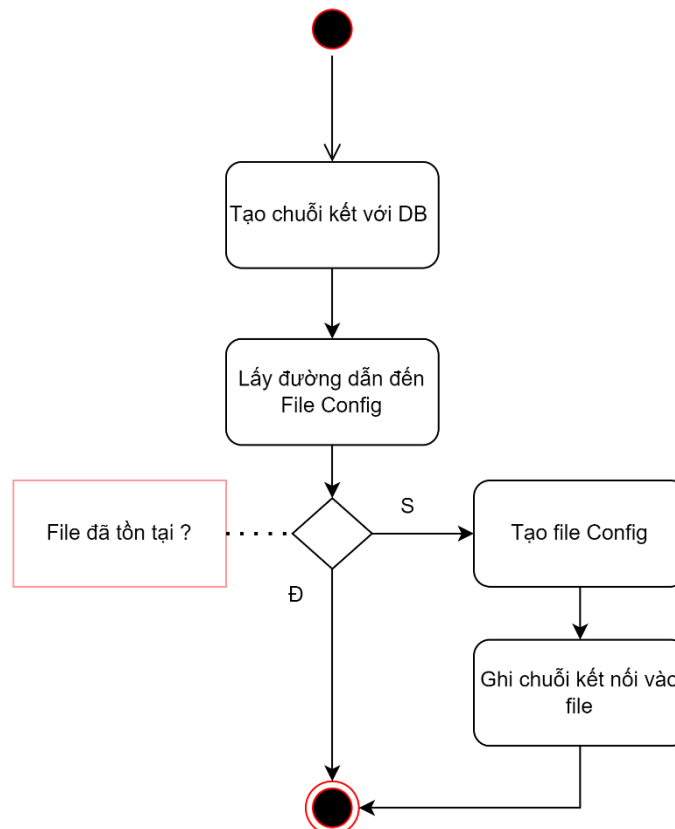
```
2023-07-24 15:32:55.567, Error: Database connection failed.
```

3.2.3 Xây dựng các lớp cơ sở

3.2.3.1 Setting class

Quản lý việc thiết lập kết nối với cơ sở dữ liệu có phương thức chính là:

- **createFile(string file, int status, string ds, string db, string user = null, string password = null):**
- Phương thức hoạt động như sau:



Hình 3.17 Activity cho tạo chuỗi kết nối

3.2.3.2 Login class

Nhằm phục vụ cho việc lưu thông tin của người dùng đang đăng nhập vào hệ thống.

Chứa các thuộc tính: Name, Role, NameID, RoleID

Chứa Phương thức:

- **getLoginDetails(string proc, Hashtable ht):** phương thức này nhận tham số đầu vào là Username và Password được truyền vào Hashtable(ht). Phương thức sẽ trả về Null nếu không tồn tại thông tin đăng nhập. Và sẽ trả về thông tin về Name, Role, NameID, RoleID nếu như tìm thấy trong cơ sở dữ liệu.

3.3 Xây dựng chương trình con Common Control UI Project

Chương trình này cung cấp lớp để xử lý các hoạt động chung về mặt giao diện trong các ứng dụng Winform. Tất cả các hàm đều là phương thức Static cho phép gọi trực tiếp từ tên lớp mà không cần khởi tạo đối tượng và lớp không chứa dữ liệu hay biến thành viên.

Dưới đây là các phương thức chính trong Class:

Bảng 3.9 Phương thức chính trong Class CCUP

Phương Thức	Mô tả
sno(DataGridView gv, string snoGV)	Duyệt qua các hàng của DataGridView và đánh số thứ tự vào cột có tên “snoGV”
showWindow(Form openWin, Form closeWin, Form MDI)	Đóng Form cũ (closeWin) Hiển thị một Form mới(openWin) Form hiển thị trong của Form chính MDI
showWindow(Form openWin, Form MDI)	Hiển thị một Form mới(openWin) Form hiển thị trong của Form chính MDI
showMessage(string msg, string type)	Hiển thị hộp thoại thông báo cho người dùng Nội dung hiển thị: msg Loại MessageBox: type bao gồm (“error”, ”success”, ”warning”)
resetEnable(Panel p)	Xóa tất cả giá trị và Enable của các component(textbox, editbox, radiobutton,...) chứa trong Panel p
resetDisable(Panel p)	Xóa tất cả giá trị và Disable của các component(textbox, editbox, radiobutton,...) chứa trong Panel p
EnableControl(Panel p)	Chỉ Enable của các component(textbox, editbox, radiobutton,...) chứa trong Panel p
DisableControl(Panel p)	Chỉ Disable của các component(textbox, editbox, radiobutton,...) chứa trong Panel p
checkControls(Panel p, int editmode = 0)	Editmode = 0: Trả về list các component đang có giá trị null và hiển thị MessageBox “Fields is not blank”

Bảng trên mô tả ngắn gọn về chức năng và cách sử dụng của các phương thức được xây dựng trong Class của chương trình Common UI Project. Các phương thức trong này hướng đến việc cung cấp các tính năng hỗ trợ chung về việc xử lý giao diện cho ứng dụng Winform. Từ đó giảm thiểu việc lặp lại code cho chương trình.

3.4 Xây dựng chương trình con Common Handle DB Project

Chương trình này chứa lớp LibCRUD cung cấp các hoạt về xử lý logic và thực hiện các phương thức để tương tác với cơ sở dữ liệu thông qua việc thực thi các Stored Procedure. Dưới đây là các phương thức chính trong Class:

- 1) **data_insert_update_delete(string proc, Hashtable ht):** Phương thức này thực hiện các thao tác Insert, Update và Delete dữ liệu vào cơ sở dữ liệu thông qua việc gọi Stored Procedure (**proc**) và truyền các tham số vào qua Hashtable (**ht**). Hàm này sẽ trả lại kết quả là số nguyên.

Ví dụ: Để thêm thông tin thuốc:

```
ht.Add("@type", cb_Type.SelectedIndex);  
ht.Add("@comp", txt_company.Text);  
ht.Add("@name", txt_medi.Text);  
LibCRUD.data_insert_update_delete("st_insertMedicine", ht);
```

- 2) **LoadDataTable(string sql):** Phương thức này truy vấn trực tiếp vào dữ liệu từ cơ sở dữ liệu thông qua câu truy vấn SQL (**sql**) và trả về một DataTable chứa kết quả là một mảng các cặp dữ liệu tương ứng.

Ví dụ: Để lấy tất cả các huyện nằm trong một tỉnh

```
LibCRUD.LoadDataTable($"SELECT * FROM district WHERE dis_proviD = {selectedProvince}");
```

- 3) **loadData(string proc, DataGridView gv, ListBox lb):** Phương thức này cũng tương tự thực hiện việc truy vấn trực tiếp dữ liệu từ cơ sở dữ liệu thông qua Stored Procedure (**proc**), sau đó đổ dữ liệu vào DataGridView (**gv**) vào các cột tương ứng thông qua tên các cột được truyền vào ListBox (**lb**).

Ví dụ: Để lấy tất cả các bệnh đổ vào các hàng trong dataGridView2 với các cột tương ứng là IDGV và diseaseGV tương ứng.

```
loadData.Items.Add(IDGV);  
loadData.Items.Add(diseaseGV);  
LibCRUD.loadData("st_getDisease", dataGridView2, loadData);
```

4) **loadList(string proc, ComboBox cb, string valueNumber, string displayMember, Hashtable ht)**: Phương thức này cũng tương tự thực hiện việc truy vấn trực tiếp dữ liệu từ cơ sở dữ liệu thông qua Stored Procedure (**proc**), nhưng data sẽ được đổ vào trong ComboBox (cb) với value là (valueNumber) và tên hiển thị là (displayMember).

Ví dụ: để lấy tất cả các loại thuốc đã tồn tại trong cơ sở dữ liệu cho vào combobox để bác sĩ có thể lựa chọn loại thuốc tương ứng.

```
LibCRUD.loadList("st_getMedicineType",cb_DescripInternal,"ID",  
"Medicine", ht1);
```

5) **getLastID(string proc)**: Phương thức này có chức năng chính là lấy ID của đối tượng nào đó ở trong database ở vị trí cuối cùng tức là mới nhất thêm vào.

Ví dụ: Để lấy thông tin ID của một đăng kí khám bệnh của bệnh nhân

```
Convert.ToInt64(LibCRUD.getLastID("st_getlastPatientID"))
```

Như vậy Với việc đóng gói các hoạt động với cơ sở dữ liệu thông qua lớp này có thể giúp việc quản lý code được tối ưu hơn, tăng tính tái sử dụng và giảm sự phức tạp trong việc lập trình

3.5 Kiểm thử phần mềm

3.5.1 Kiểm tra chức năng

3.5.1.1 Chức năng Login

Mục tiêu thử nghiệm: Đảm bảo quá trình đăng nhập hoạt động tốt.

Bảng 3.10 Kiểm thử chức năng Login

STT	Trường hợp	Thực hiện	Kết quả mong đợi	Kết quả
1	Người dùng nhập User name và Password với dữ liệu chính xác	User name: Admin Password: Admin	Đăng nhập thành công	Đạt được
2	Người dùng nhập User name và Password với dữ liệu KHÔNG chính xác		Đăng nhập thất bại	Đạt được

3.5.1.2 Chức năng lựa chọn tác vụ ở Giao diện HomScreen

Mục tiêu thử nghiệm: Đảm bảo việc lựa chọn tác vụ thực hiện đúng chức năng

Bảng 3.11 Kiểm thử chức năng lựa chọn tác vụ ở Giao diện HomeScreen

STT	Trường hợp	Thực hiện	Kết quả mong đợi	Kết quả
1	Lựa chọn các ô tác vụ và dẫn tới cửa sổ của chức năng tương ứng	Nhấn Staff button	Hiển thị Staffwindow	Đạt được
2	Lựa chọn trở về	Nhấn Back button	Hiển thị màn hình HomeScreen	Đạt được
3	Nhấp chuột vào khoảng trống		Không thực hiện hành động nào	Đạt được

3.5.1.3 Chức năng ADD

Mục tiêu thử nghiệm: Đảm bảo quá trình thêm thông tin ở các giao diện chức năng hoạt động tốt. Thử nghiệm với màn hình Medicine

Bảng 3.12 Kiểm thử chức năng ADD

STT	Trường hợp	Quá trình thực hiện	Kết quả mong đợi	Kết quả
1	Bỏ trống trường thông tin	1: Không Nhập Medicine , nhập Company: A , Type: Internal 2: Save button	2.1: Không thêm vào cơ sở dữ liệu 2.2: Warning Message : “field is blank”	Đạt được
2	Điền đủ các trường	1: Nhấn Add button 2: Nhập đầy đủ thông tin 3: Save button	3.1: Thêm thành công vào cơ sở dữ liệu 3.2: Success message: “Add medicine successfully”	Đạt được

3.5.1.4 Chức năng EDIT

Mục tiêu thử nghiệm: Đảm bảo quá trình sửa thông tin ở các giao diện chức năng hoạt động tốt. Sử dụng kiểm thử trên màn hình Medicine.

Bảng 3.13 Kiểm thử chức năng EDIT

STT	Trường hợp	Thực hiện	Kết quả mong đợi	Kết quả
3	Hiển thị đúng thông tin được select	1: Chọn một Row muốn thay đổi thông tin	1: Thông tin của thuốc đó được gán lên các field tương ứng	Đạt được
4	Bỏ trống trường thông tin	1: Chọn một Row muốn thay đổi thông tin 2: Không Nhập Medicine ,	3.1: Không thay đổi thông tin 3.2: Warning Message :	Đạt được

		3: Save button	“field is blank”	
5	Đủ trường thông tin	1: Chọn một Row muốn thay đổi thông tin 2: Nhấn Edit button 3: Không đủ thông tin 3: Save button	3.1 Thông tin của Medicine lựa chọn được thay đổi theo giá trị được nhập vào	Đạt được

3.5.1.5 Chức năng DELETE

Mục tiêu thử nghiệm: Đảm bảo quá trình thực hiện các tác vụ hoạt động chính xác. Test được thực hiện trên màn hình DiseaseScreen.

Bảng 3.14 Kiểm thử chức năng DELETE

STT	Trường hợp	Thực hiện	Kết quả mong đợi	Kết quả
6	Không chọn đối tượng delete	1: Vào màn hình và bấm Delete button	1.1: Không xóa phần tử nào 1.2: Show warning Message: “please select a row”	Đạt được
7	Đã chọn đối tượng để delete	1: Chọn một row chứa trường muốn xóa 2: Nhấn Delete button	2.1: Xóa thành công phần tử đã chọn trong cơ sở dữ liệu 2.2: Show success button “Delete successfully”	Đạt được
8	Hiển thị thông tin	1: Vào màn hình MedicineScreen 2: Nhấn View button	1: Tất cả thông tin thuốc trong cơ sở dữ liệu được hiển thị lên màn hình	Đạt được

3.5.1.6 Hiển thị thông tin ở màn BioTesScreent

Mục tiêu thử nghiệm: Để đảm bảo dữ liệu được đọc thành công từ cơ sở dữ liệu và hiển thị lên giao diện người dùng. Được kiểm thử trên màn hình BioTestScreen .

Bảng 3.15 Kiểm thử chức năng hiển thị trên màn hình BioTestScreen

STT	Trường hợp	Thực Hiện	Kết quả mong đợi	Kết quả
9	Load patiet	1: Vào màn hình BioTestScreen 2: Chọn ngày khám 3: Nhấn View button	3: Tất cả các bệnh nhân thăm khám được hiển thị trong ComboBox patient	Đạt được
10	Thêm thông tin xét nghiệm	1: Vào màn hình BioTestScreen 2: Chọn ngày khám 3: Nhấn View button 4: Chọn bệnh nhân trong danh sách đăng kí 5: Chọn Add button 6: Thêm các trường thông tin xét nghiệm 7: Nhấn Save button	4.1: Thông tin cơ bản người bệnh đăng kí được hiển thị tại các trường tương ứng. 4.2: Tất cả các trường đều bị disable 5.1: Các trường thông tin về thông tin xét nghiệm được xóa hết và enable 5.2: Các trường thông tin cơ bản của bệnh nhân vẫn là disable khôn cho phép chỉnh sửa tại màn này 7.1: Thêm thông tin thành công vào database 7.2: Show success message:”Add successfully”	Đạt được
11	Xóa thông tin xét nghiệm	1: Vào màn hình BioTestScreen 2: Chọn ngày khám	4.1: Thông tin cơ bản người bệnh đăng kí	Đạt được

		3: Nhấn View button 4: Chọn bệnh nhân trong danh sách đăng kí 5: Chọn Delete button	được hiển thị tại các trường tương ứng. 4.2: Tất cả các trường đều bị disable 5.1: Thông tin xét nghiệm của bệnh nhân sẽ được xóa 5.2: Show success message: "Delete successfully"	
--	--	---	---	--

3.6 Kết Luận

Như vậy ở chương này đã đi thiết kế hệ thống theo từng phần một khiến cho chương trình có dễ hiểu và dễ mở rộng trong tương lai hơn. Về giao diện thì hoàn thành được yêu cầu về chức năng và còn hạn chế ở sự chi tiết khi có những giá trị, hành động ngoại lệ được đưa vào. Nhưng về tổng quan thì hệ thống đã có thể vận hành thử nghiệm bên cạnh việc tiếp tục maintain về các trường hợp ngoại lệ.

CHƯƠNG 4. TỔNG KẾT VÀ PHÁT TRIỂN

Ở chương 4 sẽ đi tổng kết lại kết quả của cả một quá trình phát triển phần mềm tới thời điểm hiện tại. Còn điểm yếu, hạn chế gì ở phần mềm để có thể có hướng cải thiện và phát triển trong tương lai

4.1 Kết quả đạt được

Hệ thống quản lý thông tin của bệnh nhân thận nhân tạo gồm 1 server và các clien chạy chương trình Winfrom ban đầu đã đáp ứng được các yêu cầu đã đặt ra, đặc biệt dễ sử dụng cho các cán bộ y tế đem lại hiệu quả nhất định trong quản lý thông tin.

Phần mềm được tạo với mục tiêu đảm bảo các chức năng cần thiết trong thực tế với bệnh nhân và có thể sử dụng một cách dễ dàng cả cho người dùng và khả năng chỉnh sửa , nâng cấp phần mềm một cách nhanh chóng nhất. Trong thời gian tới, các chức năng hiện tại có thể hoàn thiện một các chĩn chu hơn.

4.2 Hạn chế của phần mềm

Phần mềm được xây dựng trong thời gian ngắn nên vẫn còn tồn tại nhiều hạn chế, nhiều điểm chưa được hoàn thiện như:

- Về giao diện chưa đảm bảo được việc bắt các tất cả các ngoại lệ mà người dùng nhập vào.
- Về việc lưu trữ thông tin xét nghiệm của bệnh nhân như: xét nghiệm máu, xét nghiệm nước tiểu có gặp vấn đề khi có nhiều loại máy cho ra các thông số khác nhau nên còn thiếu sự linh hoạt trong việc quản lý.

4.3 Hướng phát triển

Quá trình phát triển phần mềm trong lĩnh vực chăm sóc sức khỏe, đặc biệt là trong việc hỗ trợ khả năng phân tích dữ liệu của bệnh nhân điều trị thận nhân tạo đang có nhiều cơ hội.

- Về dữ liệu có thể mở rộng và tổ chức hợp lý hơn để hướng tới sự kết hợp giữa bác sĩ và công nghệ thông tin trong việc chuẩn đoán bệnh nhằm giảm bớt gánh nặng cho bác sĩ.
- Nâng cấp về khả năng thu thập và phân tích với dữ liệu lớn đây là hướng phát triển cốt yếu của dự án. Đưa ra được nhiều thông tin phân tích cho mỗi bệnh nhân dưới dạng các biểu đồ, xây dựng lên mô hình dự đoán sơ bộ tình trạng sức khỏe của bệnh nhân trước khi nhận thông tin chuẩn đoán chính thức từ bác sĩ.
- Vì liên quan đến thông tin về bệnh nhân nên phần mềm cần nâng khả năng về tính bảo mật .

KẾT LUẬN

Kết luận chung

Sau quá trình tìm hiểu, phân tích, thiết kế và triển khai xây dựng, hệ thống hỗ trợ quản lý thông tin bệnh nhân điều trị thận nhân tạo, ứng dụng hoạt động hoạt động trên Window, Server đã cơ bản được hoàn thành. Hệ thống mang lại hiệu quả nhất định trong việc quản lý ở bệnh viện hay các phòng khám

WindowApp có điểm mạnh về giao diện thân thiện với người dùng, dễ thực hiện các thao tác quản lý thiết bị. Server chạy tương đối ổn định.

TÀI LIỆU THAM KHẢO

- [1] [ER Diagram for a Hospital Management System | Vertabelo Database Modeler](#), truy cập cuối cùng 24/7/2023
- [2] [\(99+\) Hospital Database Management System\(sql project report\) | Faysal sarder - Academia.edu](#), truy cập cuối cùng ngày 24/7/2023.
- [3] [C# Windows Forms Application Tutorial with Example \(guru99.com\)](#), truy cập cuối cùng ngày 25/6/2023.
- [4] [Phân tích thiết kế hệ thống thông tin sử dụng biểu đồ UML \(Phần 1\) \(viblo.asia\)](#), truy cập cuối cùng ngày 27/5/2023.
- [5] [SQL Server Tutorial – The Practical SQL Server Tutorial](#), truy cập cuối cùng ngày 2/8/2023.
- [6] [newonme.drawio - draw.io \(diagrams.net\)](#), truy cập lần cuối cùng ngày 30/7/2023
- [7] Dương Quang Thiện, *tập 1 – C# căn bản*, 2005.
- [8] Dương Quang Thiện, *tập 2 – C# và .NET Framework*, 2005.
- [9] Nguyễn Ngọc Bình Phương và Thái Thanh Phong, *Các giải pháp lập trình C#*