TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––

Ảnh có chứa văn bản

Mô tả được tạo tự động

**BÁO CÁO THỰC TẬP NGHỀ NGHIỆP**

**NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ**

**CHUYÊN NGÀNH QUẢN TRỊ HỆ THỐNG THÔNG TIN**

**THỰC HIỆN KIỂM THỬ PHẦN MỀM QUẢN LÝ BỆNH VIỆN ĐA KHOA GIA ĐÌNH ĐÀ NẴNG**

Sinh viên thực hiện : Nguyễn Tống Khánh Linh

Lớp : 45K21.1

Đơn vị thực tập : Bệnh Viện Đa Khoa Gia Đình ĐN

Cán bộ hướng dẫn : Nguyễn Thị Thanh Mai

Giảng viên hướng dẫn : Ths. Cao Thị Nhâm

**Đà Nẵng, 8/2022**

**NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP**

Họ và tên sinh viên:

Lớp: Khoa: Trường:

Thực tập từ ngày: …./……/ 2022 đến ngày: ........./ .……./ 2022

Tại:

Địa chỉ:

Sau quá trình thực tập tại đơn vị của sinh viên, chúng tôi có một số nhận xét, đánh giá như sau:

**1. Về thái độ, ý thức, đạo đức, kỷ luật**

**2. Kiến thức chuyên môn**

**3. Khả năng hòa nhập và thích nghi với công việc**

**4. Trách nhiệm, sáng tạo trong công việc**

**5. Các nhận xét khác**

**Đánh giá chung:**

**Điểm:**

……….., ngày .......tháng ......năm 2022

**Xác nhận của đơn vị thực tập**

# LỜI CẢM ƠN

# LỜI CAM ĐOAN

Quy định:

Soạn thảo trên trang A4 (trang dọc, lề trái: 3.5cm; trên, phải, dưới: 2.5 cm), Font Times New Roman, canh đều 2 bên, size 13, cách dòng 1.5, cách đoạn trên 6pt, cách đoạn dưới 3pt, hàng đâu tiên lùi vào 1.27 cm; hình và bảng soạn theo caption, chèn trích dẫn chéo (Cross-reference) cho bảng và hình; các danh mục hình, bảng, mục lục làm tự động; đánh số trang như file mẫu (bìa không có số trang, danh mục + mục lục số trang theo i, ii, iii…, nội dung chính theo 1,2,3…)

# MỤC LỤC

[NHẬN XÉT CỦA DOANH NGHIỆP THỰC TẬP ii](#_Toc98339732)

[LỜI CẢM ƠN iii](#_Toc98339733)

[LỜI CAM ĐOAN iv](#_Toc98339734)

[MỤC LỤC vi](#_Toc98339735)

[DANH MỤC HÌNH ẢNH viii](#_Toc98339736)

[DANH MỤC BẢNG BIỂU ix](#_Toc98339737)

[DANH MỤC CÁC TỪ VIẾT TẮT x](#_Toc98339738)

[LỜI MỞ ĐẦU 1](#_Toc98339739)

[CHƯƠNG 1. TỔNG QUAN … 2](#_Toc98339740)

[1.1. Mục 1.1 2](#_Toc98339741)

[1.1.1. Mục 1.1.1 2](#_Toc98339742)

[1.1.2. Mục 1.1.2 2](#_Toc98339743)

[1.2. Mục 1.2 2](#_Toc98339744)

[CHƯƠNG 2. LÝ THUYẾT 3](#_Toc98339745)

[2.1. Mục 2.1 3](#_Toc98339746)

[2.1.1. Mục 2.1.1 3](#_Toc98339747)

[2.1.2. Mục 2.1.2 3](#_Toc98339748)

[2.2. Mục 2.2 3](#_Toc98339749)

[CHƯƠNG 3. TRIỂN KHAI … 4](#_Toc98339750)

[3.1. Mục 3.1 4](#_Toc98339751)

[3.1.1. Mục 3.1.1 4](#_Toc98339752)

[3.1.2. Mục 3.1.2 4](#_Toc98339753)

[3.2. Mục 3.2 4](#_Toc98339754)

[CHƯƠNG 4. KẾT QUẢ 5](#_Toc98339755)

[4.1. Mục 4.1… 5](#_Toc98339756)

[4.2. Mục 4.2… 5](#_Toc98339757)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 6](#_Toc98339758)

[TÀI LIỆU THAM KHẢO 7](#_Toc98339759)

[PHỤ LỤC 8](#_Toc98339760)

*(Mục lục này chỉ là ví dụ)*

# DANH MỤC HÌNH ẢNH

[Hình 1.1 Ngôn ngữ lập trình Python 2](#_Toc98336120)

[Hình 4.1 Kiến trúc của mô hình 5](#_Toc98336121)

# DANH MỤC BẢNG BIỂU

[Bảng 4.1 Kiến trúc 5](#_Toc74235471)

# DANH MỤC CÁC TỪ VIẾT TẮT

**AI** : Artificial Intelligence

**…**

# LỜI MỞ ĐẦU

1. **Mục tiêu nghiên cứu của đề tài**

* Đề tài này nghiên cứu …

1. **Nhiệm vụ của đề tài**

* Nghiên cứu ...
* ...

1. **Phương pháp nghiên cứu**

* ...

1. **Đối tượng và phạm vi nghiên cứu**

* ...

1. **Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, x chương nội dung và phần kết luận...

* Mở đầu
* **Chương 1**:
* **Chương 2**:
* **Chương 3**:
* **Chương 4**:
* Kết luận và hướng phát triển

# CƠ SỞ LÝ THUYẾT

## Tổng quan các khái niệm cơ bản về kiểm thử phần mềm

### Kiểm thử phần mềm là gì?

Kiểm thử phần mềm là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng của sản phẩm hoặc dịch vụ được kiểm thử. Cung cấp cho doanh nghiệp một quan điểm, cách nhìn độc lập về phần mềm để từ đó cho phép đánh giá và thấu hiểu rủi ro trong quá trình triển khai phần mềm.

### Mục đích của kiểm thử phần mềm

* Tìm các bug phát sinh do dev tạo ra khi code và ngăn ngừa lỗi
* Phê chuẩn và xác minh một ứng dụng/ sản phẩm:
  + Hoàn thành công việc đúng như kì vọng
  + Đáp ứng được nhu cầu của các bên liên quan:
    - Yêu cầu của doanh nghiệp và người dùng
    - Đáp ứng BRS ( Business Requirement Specification) – đặc tả yêu cầu kinh doanh và SRS( System Requirement Specification) – đặc tả yêu cầu hệ thống.
    - Đạt được sự tín nhiệm của khách hàng bằng cách cung cấp cho họ một sản phẩm chất lượng.

### Vai trò của kiểm thử phần mềm

Kiểm thử phần mềm có vai trò quan trọng trong sự phát triển phần mềm, Kiểm thử bao phủ các lĩnh vực khác nhau như: chức năng các ứng dụng, khả năng tương thích của các ứng dụng với các hệ điều hành, phần cứng và các loại khác nhau của trình duyệt để đảm bảo rằng sẽ không có bất kì vấn đề trục trặc nào.

Kiểm tra xác nhận rằng hệ thống đáp ứng các yêu cầu khác nhau bao gồm: chức năng, hiệu suất, độ tin cậy, an toàn, khả năng sử dụng để có thể tiết kiệm thời gian, tiết kiệm chi phí và đem lại sự hài lòng của khách hàng.

### Người thực hiện kiểm thử (test)

Điều này sẽ phụ thuộc vào quy trình của các bên liên quan đến dự án. Trong ngành công nghiệp phần mềm, thì sẽ có 1 team chuyên chịu trách nhiệm về việc đánh giá phần mềm phát triển với yêu cầu đã được chỉ định – gọi là Tester. Tại bệnh viện thì đội ngũ Tester ấy sẽ được Test trong một chức năng của phần mềm tại Bệnh viện. Vì thế, đối với vấn đề Tester thì có thể là:

* Solfware Tester – nhân viên kiểm thử phần mềm
* Solfware Developer – nhân viên phát triển phần mềm
* Leader hoặc manager dự án
* Product Owner – người sở hữu sản phẩm
* User – người dùng cuối

### Thời điểm kiểm thử phần mềm

Tuỳ vào từng mô hình phát triển phần mềm mà thời gian thực hiện test là khác nhau. Thực hiện kiểm thử thì càng sớm càng tốt. Code xong thì Dev sẽ build và bàn giao cho Tester thực hiện test.

Thực tế thì tester tham gia sớm hơn, chỉ cần có tài liệu Đặc tả Yêu cầu/ Nghiệp vụ là tester thực hiện tìm hiểu nghiệp vụ dự án và thực hiện viết Testcase

## Quy trình kiểm thử phần mềm

### Lập kế hoạch kiểm thử (test plan)

Kế hoạch kiểm thử là một tài liệu mô tả các mục tiêu, phạm vị, phương pháp, tiếp cận và tập trung vào nỗ lực kiểm thử phần mềm.

* Cấu trúc chung của một test plan:
  + Tên Project
  + Danh sách module cần test
  + Ngày bắt đầu, ngày kết thúc
  + Nhân sự tham gia ( Tham gia cuộc họp với khách hàng)
  + Kế hoạch thực hiện ( sử dụng Excel lập kế hoạch)

Khi viết test plan thì nó sẽ giúp chúng ta định hướng suy nghĩ, là phương tiện giao tiếp của nhóm test với đội dự án và công cụ giao việc trong nhóm test.

### Viết test case

Testcase là một tập hợp các thông số đầu vào kiểm thử, điều kiện thực thi và kết quả mong đợi được phát triển cho một mục tiêu cụ thể. Một trường hợp kiểm thử có thể chỉ đơn giản là một câu hỏi cho chương trình. ***(Theo: Wikipedia.vn)***

Một trường hợp kiểm thử có thể có các phần đặc thù khác nhau như: Mã Testcase, tên test case, Sub-items (mục con của mục mình test), Steps to excute (các bước thực hiện), Expected out put (KQ mong muốn) và kết quả thực tế.

Nó bao gồm 3 bước cơ bản:

* Mô tả: Đặc tả các điều kiện cần có để tiến hành kiểm tra
* Nhập: Đặc tả đối tượng hoặc dữ liệu cần thiết để thực hiện kiểm tra
* Kết quả mong muốn: Trả về từ đối tượng kiểm tra.

### Thực hành test

Việc test plan và viết testcase đã xong thì thực hành test sẽ được diễn ra dựa trên testcase đã viết. Đánh giá kết quả kiểm thử cho các trường hợp kiểm thử. So sánh kết quả thực tế khi thực thi với kết quả mong muốn.

### Viết báo cáo kiểm thử (test report)

Báo cáo kiếm thử thể hiện tiến độ kiểm thử, tiến độ sửa lỗi và số lượng lỗi được tìm thấy hay còn tồn của dự án. Nó còn đánh giá và giám sát xem dự án có kịp tiến độ để bàn giao cho khách hay không.

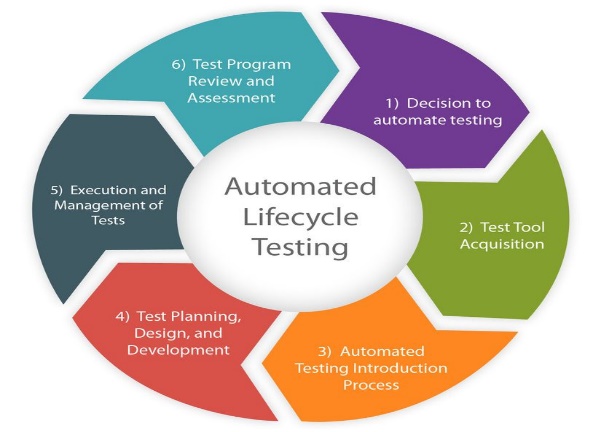
## Các loại kiểm thử phần mềm ( Software Testing)

### Kiểm thử thủ công: Manual testing



* Manual Testing không sử dụng bất kì công cụ tự động hoặc script
* Đảm nhận vai trò của người dùng cuối, xác định hành vi và lỗi không mong muốn

### Kiểm thử tự động: Automation testing



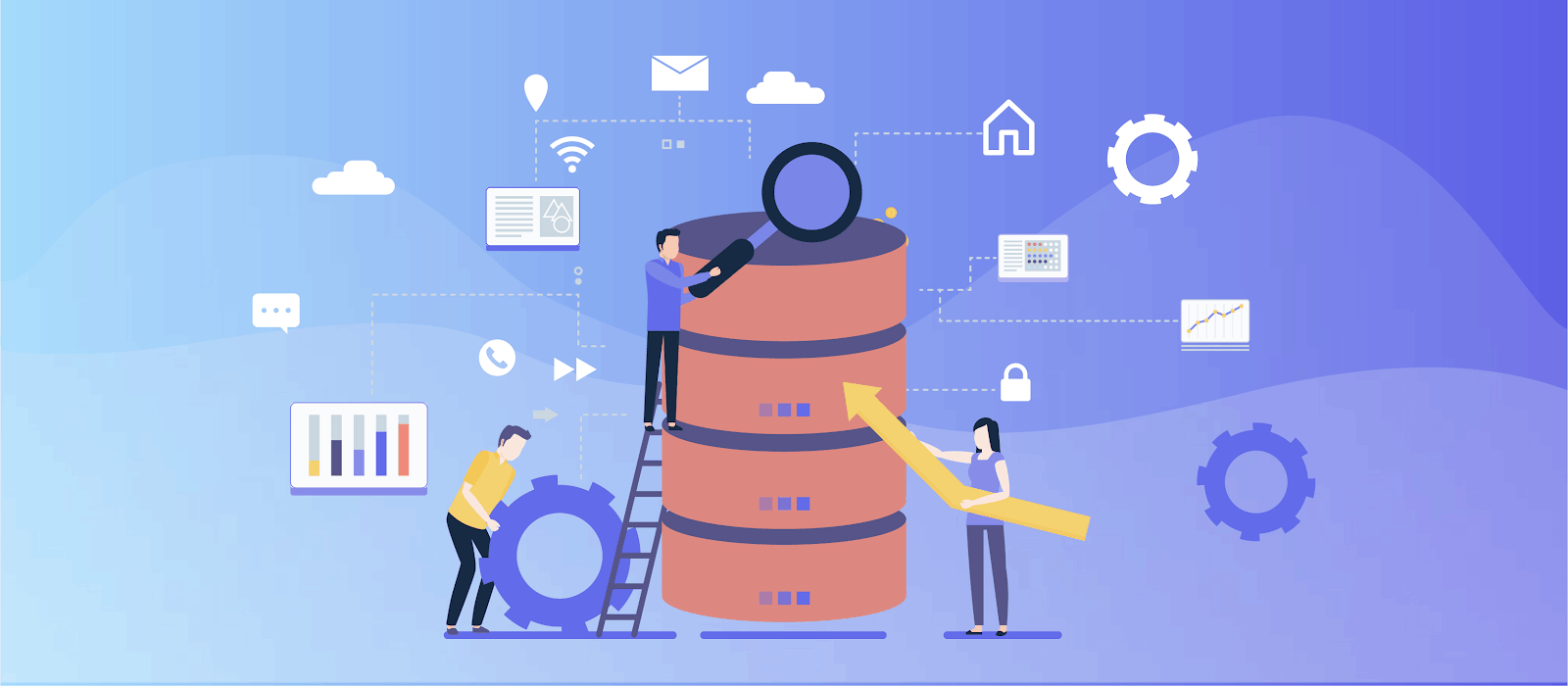
* Automation testing liên quan đến tự động hoá quy trình thủ công, thực hiện nhanh chóng hơn
* Nó làm tăng phạm vi kiểm tra, cải thiện độ chính xác và tiết kiệm thời gian và tiền bạc so với kiểm tra thủ công

## Giới thiệu về SQL

### Tổng quan về Database và sự ra đời của SQL

#### Tổng quan về Database

Data là thông tin dữ liệu gắn kèm, xung quanh 1 đối tượng nào đó, ví dụ như: data cá nhân bao gồm: họ tên, giới tính,….. Hoặc đối với data của sinh viên bao gồm: họ tên, quê quán, mã sinh viên



Database (cơ sở dữ liệu) là một tập hợp những data (dữ liệu) có liên quan với nhau. Nó được duy trì dưới dạng một tập hợp các tập tin trong hệ điều hành được lưu trữ trong các hệ quản trị cơ sở dữ liệu ***(Theo Wikipedia.vn)***

Database là kho lưu trữ dữ liệu, sẽ hỗ trợ truy xuất và quản lý dữ liệu được một cách dễ dàng, cách thức phụ thuộc vào module quan hệ.

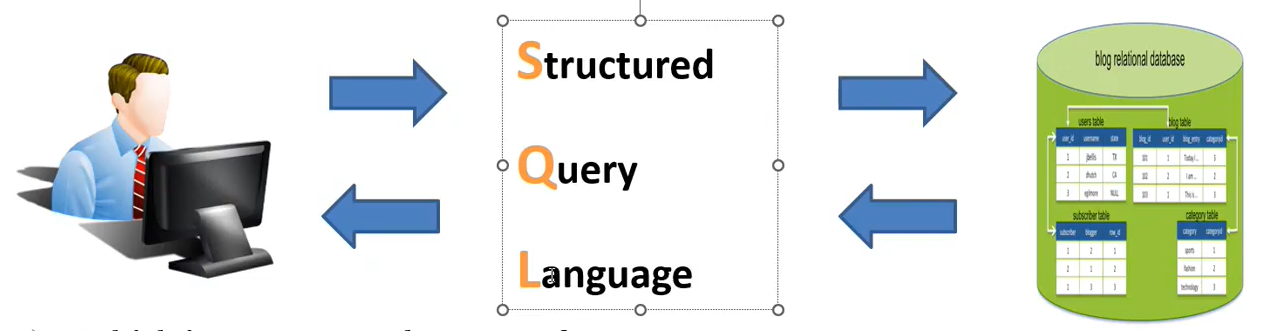
Relational Database (RDB) được lưu trữ ở dạng bảng, giữa các bảng thì sẽ có quan hệ tham chiếu với nhau. Đối với cách thức lưu trữ dữ liệu thì sẽ có 4 cách: Phân cấp, network, quan hệ, hướng đối tượng.

* Quan hệ: Cách thức lưu trữ dữ liệu này bao gồm 12 rules nó sẽ đưa ra khái niệm sơ khai mô hình cách thức lưu trữ database theo dạng quan hệ.
* Hướng đối tượng: Thường sẽ dùng NoSQL để truy cập và quản lý dữ liệu.

Relational Database Management System(RDBMS) được gọi là hệ quản trị cơ sở dữ liệu, là các công cụ, các tool hỗ trợ quản lý và lưu trữ RDB: SQL Server, MS Access, MySQL, Oracle DB, IBM DB2,…

#### Sự ra đời của SQL

Vậy Database làm như thế nào để truy xuất và quản lý dữ liệu thì SQL ra đời phục vụ điều đó. SQL hay còn gọi là Structured Query Language đơn thuần là ngôn ngữ giao tiếp giữa user và RDB.



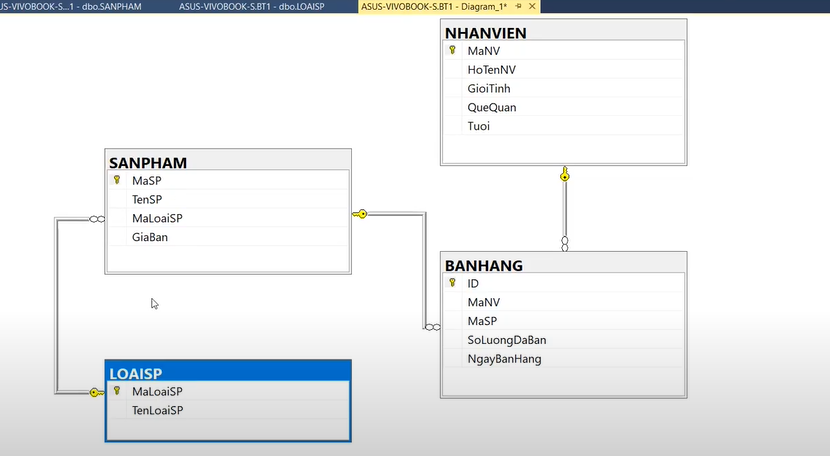
#### Các nhóm lệnh cơ bản

|  |  |  |
| --- | --- | --- |
| Nhóm lệnh | Lệnh | Định nghĩa |
| Data Control Language  DCL | Grant  Revole | Ngôn ngữ điều khiển dữ liệu cấp quyền hoặc huỷ quyền của người dùng trên đối tượng của cơ sở dữ liệu |
| Data Manipulation Language  DML | Delete  Insert  Select  Update | Ngôn ngữ thao tác dữ liệu dùng để thay đổi dữ liệu có trong các bảng của cơ sở dữ liệu |
| Data Definition Language  DDL | Create  Alter  Drop | Ngôn ngữ định nghĩa dữ liệu tạo cấu trúc, tạo mối quan hệ ràng buộc giữa các bảng với nhau. |

#### Ràng buộc dữ liệu

Các ràng buộc dữ liệu dùng để kiểm tra sự đúng đắn và tính toàn vẹn của dữ liệu. Ràng buộc gồm các loại sau:

* PK = Primary Key = Khoá định danh = Khoá chính:
  + Không được phép trùng nhau
  + Không được phép NULL
  + Có thể tạo thành từ 1 hoặc nhiều cột (sẽ có trường hợp trong 1 bảng có nhiều cột kết hợp với nhau để tạo thành khoá chính)
  + Một bảng chỉ có duy nhất 1 khoá chính
* FK = Foreign Key = Khoá tham chiếu = Khoá ngoại:
  + Nhập data có tồn tại ở khoá chính mà nó đang tham chiếu
  + Có thể trùng lặp
  + Được phép NULL (nếu cột đó đang khai báo là Allow Null)
  + Một bảng chỉ có thể có nhiều khoá ngoại



* CHECK: Dùng để kiểm tra dữ liệu đầu vào của 1 cột xác định

#### Các kiểu dữ liệu cơ bản

Trong SQL thì có 3 loại dữ liệu chính bao gồm: Loại kiểu số (dùng trong mục đích tính toán), ký tự (biểu diễn thông tin) và thời gian (ngày, giờ)

##### String data

* char(length): kiểu ký tự với độ dài cố đinh, lưu trữ kí tự dưới dạng ASCII
* varchar(length): kiểu ký tự với độ dài thay đổi, có thể lấy dữ liệu được nằm bên trong nó, lưu trữ kí tự dưới dạng ASCII
* nchar(length): kiểu ký tự với độ dài cố định và có thể lưu trữ ở các ký tự Unicode
* nvarchar(length): kiểu ký tự với độ dài thay đổi và có thể lưu trữ ở các ký tự Unicode

\*Với length là số lượng ký tự tối đa.

* Trường hợp là ký tự bình thường(char,varchar) thì phải nằm trong khoảng(1,8000)
* Trường hợp là ký tự Unicode(nchar,nvarchar) thì phải nằm trong khoảng (1,4000)

##### Numeric data

* Kiểu số nguyên:

|  |  |  |
| --- | --- | --- |
| Data Type | Size | Range of values |
| Bigint | 8 bytes | -2^63 to 2^63-1 |
| Int | 4 bytes | -2^31 to 2^31-1 |
| Smallint | 2 bytes | -2^15 to 2^15-1 |
| Tinyint | 1 bytes | 0 -> 255 |
| Bit | 1 bytes | 1. và 1 |

* Kiểu số thực(Exact Numbers/số chính xác):

|  |  |
| --- | --- |
| Data Type | Size |
| Decimal(p,s) | 5 -> 17 bytes |
| Numeric(p,s) | 5 -> 17 bytes |
| Smallmoney | 4 bytes |
| Money | 8 bytes |

* + p: số chữ số tối đa của cả phần nguyên và phần thập phân
  + s: số chữ số tối đa của phần thập phân
* Kiểu số thực (gần đúng):

|  |  |
| --- | --- |
| Data Type | Size |
| Float | 8 bytes |
| Real | 4 bytes |

##### DateTime data

|  |  |
| --- | --- |
| Data Type | Mô tả |
| Date | Ngày – tháng – năm |
| Datetime | Ngày – tháng – năm – giờ - phút - giây. Có thêm 3 kí tự thập phân đằng sau giây |
| Datetime2 | (tương ứng với Timestamp) Ngày – tháng – năm – giờ - phút – giây. Giây của nó tới nanos |
| Datetimeoffset | Ngoài ngày giờ ra còn có múi giờ |
| Smalldate time | Ngày – tháng – năm – giờ - phút – giây và không có kí tự thập phân |
| Time | Chỉ có giờ tới nanos |

#### Các phép toán tử

##### Toán tử tính toán

Các phép toán học bao gồm: +, -, \*, /. Các phép toán này thường được dùng ở mệnh đề SELECT, WHERE, HAVING.

##### Toán tử logic

|  |  |
| --- | --- |
| Toán tử | Mô tả |
| AND | Toán tử AND cho phép nhiều điều kiện cùng tồn tại |
| ALL | Toán tử ALL được sử dụng để so sánh một giá trị với tất cả các giá trị trong tập hợp giá trị khác. |
| OR | Toán tử OR thường được sử dụng để nối nhiều điều kiện |

##### Toán tử so sánh

Phép so sánh gồm: >, >=, <, <=, = và <>(!=). Các phép toán này thường được dùng ở mệnh đề WHERE hoặc HAVING, dùng để giới hạn kết quả trả về. Kết quả trả về phép so sánh là True/False.

### Các lệnh DDL

#### Lệnh tạo bảng (CREATE TABLE)

##### Quy tắc đặt tên bảng:

* Không quá 50 ký tự
* Tên bảng bắt đầu bằng chữ cái
* Tên bảng/cột không dùng những cụm từ đã có sẵn trong SQL Server
* Tên cột phải duy nhất trong bảng

##### Cú pháp tạo Database:

Create Database mydb5

Use mydb5;

##### Cú pháp tạo Table: Trực tiếp lưu trữ thông tin

Create Table tintuc(

tieude varchar(50),

noidung varchar(500))

#### Lệnh xoá bảng (DROP TABLE)

Drop Table tintuc

#### Lệnh sửa bảng (ALTER TABLE)

##### Sửa lại và thêm cột

Alter Table tintuc

Add nguoidang varchar(30);

##### Sửa lại và xoá cột

Alter Table tintuc

Drop column nguoidang varchar(30);

### Các lệnh DML

#### Lệnh Insert

Cú pháp:

Insert into tên\_bảng (cột1, cột2, ...) values (gt1, gt2, ...)

#### Lệnh Update

Cú pháp:

Update tên\_bảng

Set cột 1=gt1, cột 2=gt2

Where điều\_kiện

#### Lệnh Delete

Cú pháp:

Delete From tên\_bảng

Where điều\_kiện

#### Lệnh Select

Một câu truy vấn bắt buộc phải có mệnh đề Select...From. Các mệnh đề khác là tuỳ chọn.

Cú pháp tổng quát:

Select [ALL/DISTINCT] danh\_sách\_cột

From {table\_name | view\_name}

[Where điều\_kiện]

[Group by danh\_sách\_cột\_1]

[Having điều\_kiện\_lọc]

[Order By danh\_sách\_cột\_2 [ASC | DESC]]

### Các hàm gộp/nhóm khác

|  |  |
| --- | --- |
| Tên hàm | Mô tả |
| AVG | Tính giá trị trung bình, chỉ áp dụng cho cột có kiểu số |
| MIN | Tìm giá trị nhỏ nhất đối với dữ liệu số,date  Tím giá trị nhỏ nhất theo thứ tự trong từ điển đối với dữ liệu chuỗi |
| MAX | Ngược lại với hàm (MIN) |
| COUNT(\*) | Đếm số bản ghi |
| COUNT | Đếm các giá trị khác NULL |

***Lưu ý:*** Hàm này có thể tác động lên toàn bộ dữ liệu hoặc các nhóm dữ liệu. Các hàm này chỉ dùng ở mệnh đề Select hoặc Having.

# TỔNG QUAN GIỚI THIỆU VỀ BỆNH VIỆN ĐA KHOA GIA ĐÌNH ĐÀ NẴNG

## Tổng quan về bệnh viện Đa khoa Gia Đình Đà Nẵng

Kế thừa phong cách phục vụ, sự tin cậy từ hơn 1 vạn thân chủ của Trung tâm Bác sĩ Gia Đình, FAMILY nay vươn mình thành Bệnh viện Đa khoa Gia Đình với quy mô 250 giường bệnh đầy đủ các chuyên khoa sâu.

Ảnh có chứa văn bản, bầu trời, ngoài trời, con đường

Mô tả được tạo tự động

Bước chân vào bệnh viện, các bạn sẽ được đội ngũ chăm sóc khách hàng hướng dẫn cặn kẽ các quy trình khám bệnh, thủ tục và hồ sơ khám nhanh chóng. FAMILY được bố trí 3 tầng khám bệnh từ tầng 2,3,4 với hệ thống lễ tân và quầy thuốc lễ tân từng tầng nhằm giảm thiểu tối đa thời gian chờ đợi và hoàn tất thủ tục khám cho bệnh nhân. Từ tầng 5 là khu phòng mổ và hậu phẫu được quản lý nghiêm ngặt và máy móc hiện đại. Từ tầng 6 đến tầng 9 là khu nội trú được thiết kế khang trang, sạch sẽ, thoải mái cho người bệnh.

FAMILY có một đội ngũ hỗ trợ ngoại ngữ luôn sẵn sàng đáp ứng nhu cầu của khách hàng trong và ngoài nước, cùng những dịch vụ cao cấp ưu đãi hơn dành cho khách hàng.

Ảnh có chứa trong nhà, sàn, tường, phòng bệnh viện

Mô tả được tạo tự động

Đặc biệt, toàn bộ thông tin thân chủ đều được nhập và lưu trữ vĩnh viễn vào hệ thống hồ sơ bệnh án điện tử. Toàn bộ quá trình và lịch sử khám, điều trị của bệnh nhân sẽ luôn là dữ liệu tuyệt mật chỉ dành cho việc hỗ trợ điều trị tại FAMILY hay khi bệnh nhân yêu cầu.

## Tầm nhìn, sứ mệnh

Ảnh có chứa văn bản, bảng trắng

Mô tả được tạo tự động

* Sứ mệnh:
  + Với khách hàng:

“Thân thiết như người nhà” luôn là tiêu chí hàng đầu của FAMILY.

* + Với nhân viên:

Tạo dựng môi trường làm việc thân thiện, cải tiến liên tục dựa trên sự phát triển độc lập của từng cá nhân và khả năng phối hợp làm việc theo nhóm, theo quy trình của bệnh viện.

* + Với cộng đồng xã hội:

Chung tay vì cộng đồng là cam kết của FAMILY đến với người dân.

* Tầm nhìn:

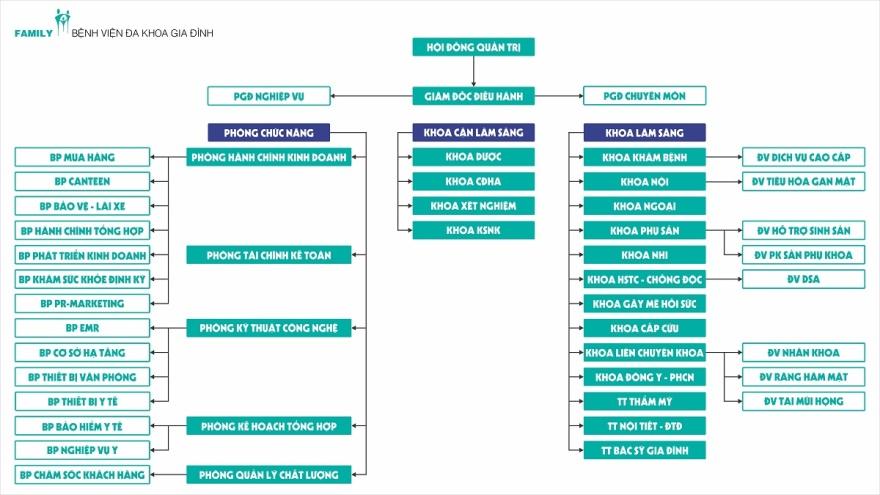
Trở thành Bệnh Viện tư cung cấp dịch vụ chăm sóc Y tế tốt nhất tại miền Trung và Tây Nguyên.

* Giá trị cốt lõi:

Phát triển dựa trên mô hình “Bác Sỹ Gia Đình”, giờ đây vẫn luôn là tôn chỉ và giá trị cốt lõi mà mỗi nhân viên y tế tại FAMILY hướng tới.

## Sơ đồ tổ chức

Bệnh viện bao gồm 12 Khoa với việc ứng dụng công nghệ thông tin, bệnh án điện tử trong quá trình khám và điều trị thì tại các khoa nhanh gọn, giảm thiểu thời gian chờ đợi của bệnh nhân.

****

# PHÂN TÍCH PHẦN MỀM “QUẢN LÝ BỆNH VIỆN”

## Tổng quan về phần mềm

### Giới thiệu về phần mềm: Quản lý bệnh viện

Có thể nói khi công nghệ chưa phát triển thì việc quản lý sẽ cực kì mất nhiều thời gian bởi vô số nhiều giấy tờ khác nhau sau đó lưu lại trong kho. Vì thế, việc phát triển phần mềm nhằm hỗ trợ cho công tác quản lý bệnh viện một cách chuyên nghiệp hơn và tiết kiệm thời gian, kiểm kể một cách tự động và dễ dàng.

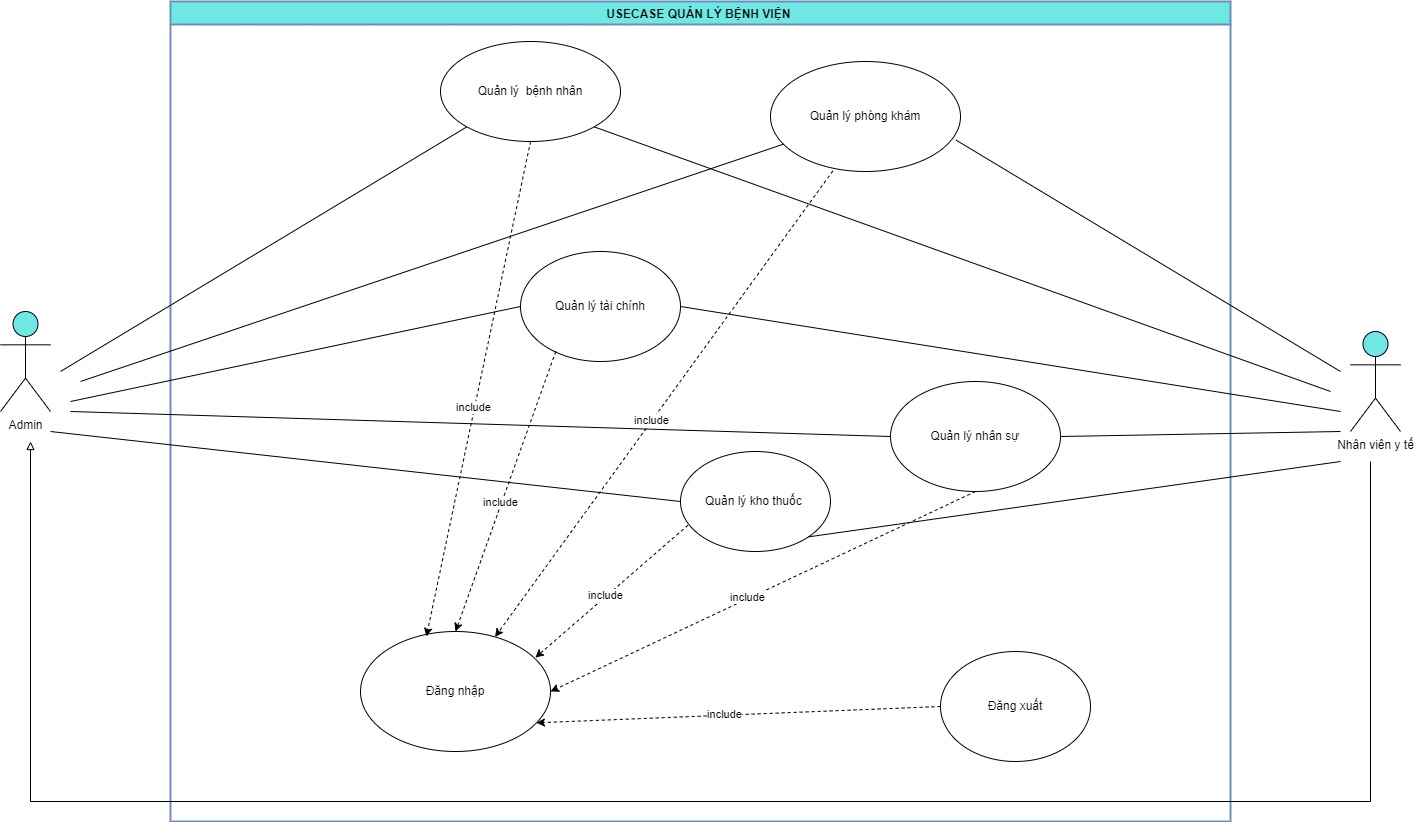
Phần mềm trên giúp nhân viên y tế có thể giám sát, xử lý dữ liệu nhanh, dễ thao tác và giúp Bệnh viện sẽ quản lý được các thông số một cách chính xác. Ngoài ra, giúp các Khoa sẽ quản lý được mỗi danh mục của Phòng ban, Khoa của mình.

### Các chức năng của phần mềm

* Hệ thống có những chức năng sau:
  + Đăng nhập
  + Quản lý bệnh nhân
  + Quản lý tài chính
  + Quản lý nhân sự
  + Quản lý phòng khám
  + Quản lý kho thuốc
  + Đăng xuất

### Sơ đồ hệ thống usecase tổng quát

* Các tác nhân chính của hệ thống bao gồm:
  + Admin (quản trị viên)
  + Nhân viên y tế



### Vai trò của từng tác nhân

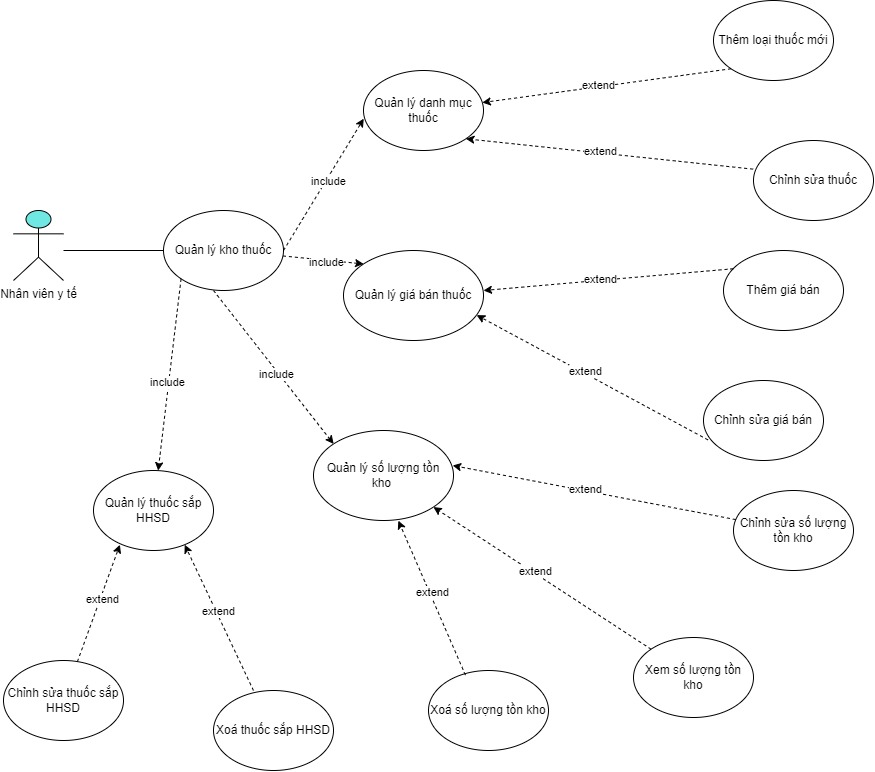
|  |  |  |
| --- | --- | --- |
| STT | Tác nhân | Chức năng |
| 1 | Admin | Tác nhân thực hiện quản lý, giám sát và nắm bắt từng yêu cầu từ phía những đối tượng người dùng khác.  Admin có thể thực hiện toàn quyền đối với hệ thống, các chức năng mà Admin có thể thực hiện bao gồm:   * Quản lý bệnh nhân * Quản lý tài chính * Quản lý nhân sự * Quản lý phòng khám * Quản lý kho thuốc * Đăng nhập/ Đăng xuất |
| 2 | Nhân viên y tế | Nhân viên y tế nắm những chức năng liên quan đến bộ phận của mình và danh sách bệnh nhân trong khoa của họ.  Ngoại trừ, một số quyền liên quan đến nhân viên y tế thì mỗi nhân viên y tế có đầy đủ các quyền của Admin, bao gồm các chức năng như sau:   * Quản lý bệnh nhân * Quản lý tài chính * Quản lý nhân sự * Quản lý phòng khám * Quản lý kho thuốc * Đăng nhập/ Đăng xuất |

Trong giai đoạn này, hệ thống tập trung phát triển các chức năng liên quan của Nhân viên y tế - đối tượng được coi là sử dụng hệ thống nhiều nhất. Tham gia vào dự án em tiến hành lập kế hoạch và thực hiện kiểm thử cho các use case sau:

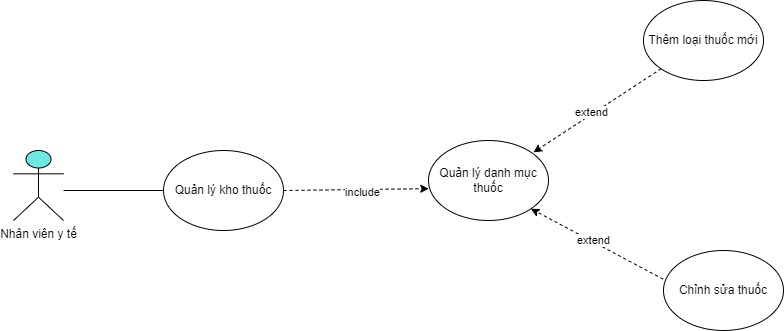
* Use case “Danh mục thuốc” – thuộc use case “Quản lý kho thuốc”
* Dưới đây là nôi dung phân tích nghiệp vụ cho use case đã hoàn thành kiểm thử.

## Phân tích Use case “Danh mục thuốc”

### Sơ đồ use case tổng quát cho chức năng “Quản lý kho thuốc”

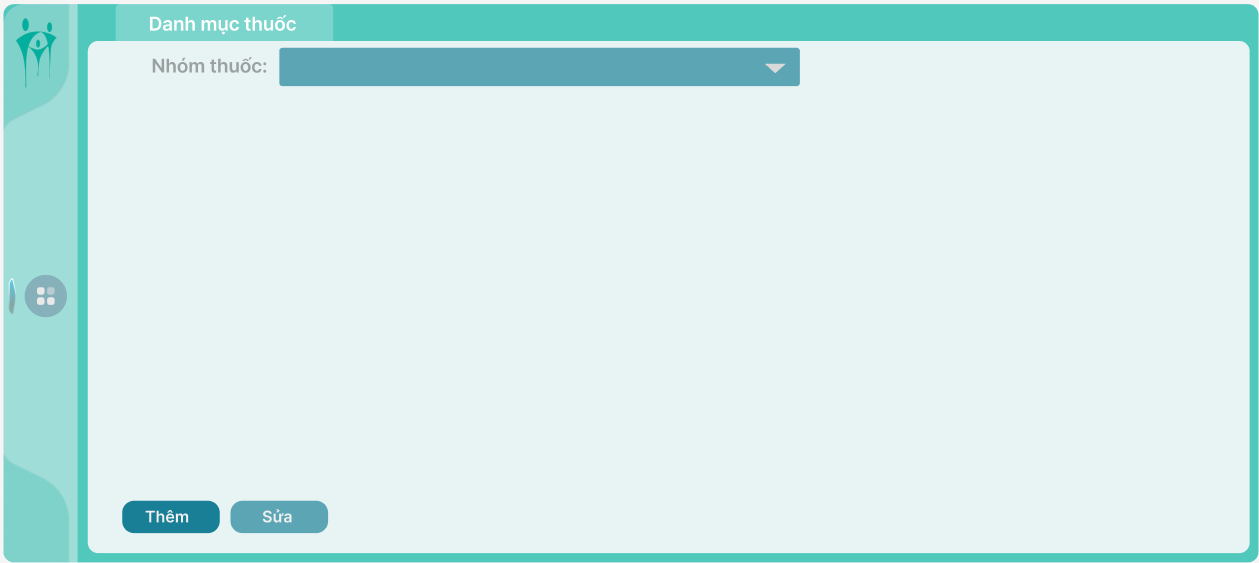


### Sơ đồ use case chi tiết cho chức năng “Danh mục thuốc”



### Phân tích use case “Thêm”

* **Tác nhân:** Nhân viên y tế
* **Điều kiện tiên quyết:** Đăng nhập vào hệ thống thành công
* **Mô tả khái quát:** Use case này được thực hiện nhằm thêm một loại thuốc và quản lý kho thuốc đó.
* **Mô tả chi tiết:** 
  + **Bước 1:** Nhân viên y tế muốn thêm loại thuốc mới, thực hiện click vào button “Thêm” tại màn hình “Danh mục thuốc”



Hình 3.1. Màn hình giao diện tại Danh mục thuốc

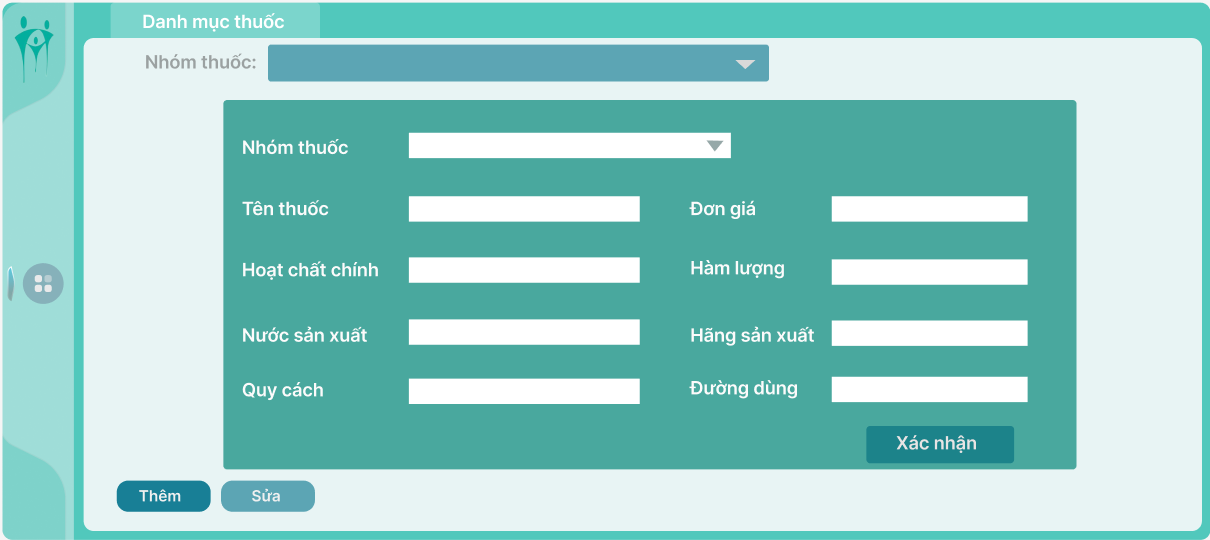
* + **Bước 2:** Hệ thống sẽ hiển thị màn hình, tại đây nhân viên y tế sẽ cập nhật các thông tin của loại thuốc bao gồm:

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích |
| Nhóm thuốc | x | User chọn nhóm thuốc phù hợp |
| Tên thuốc | x | User thêm tên thuốc |
| Hoạt chất chính |  | User thêm hoạt chất chính phù hợp của tên thuốc cần thêm |
| Nước sản xuất |  | User thêm tên nước sản xuất của tên thuốc |
| Quy cách |  | User thêm quy cách của tên thuốc |
| Hãng sản xuất |  | User thêm hãng sản xuất của tên thuốc |
| Đơn giá | x | User thêm đơn giá của tên thuốc |
| Hàm lượng | x | User thêm hàm lượng uống phù hợp để dễ dàng đưa đến thông tin cho bệnh nhân |
| Đường dùng | x | User thêm đường dùng của thuốc để hướng dẫn bệnh nhân như: uống, tiêm,.. |

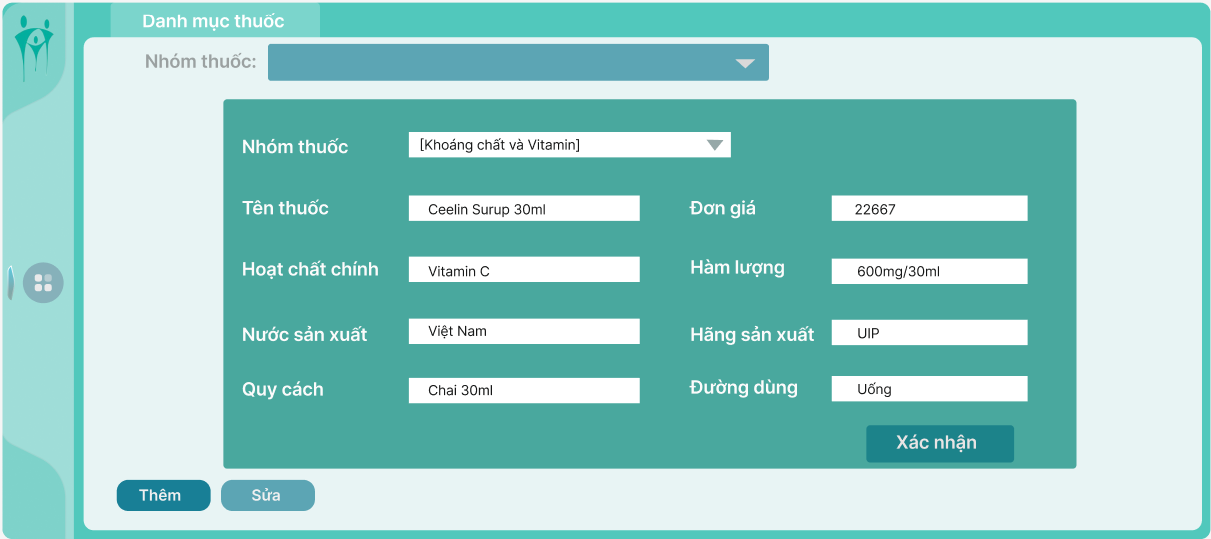
* + **Bước 3:** Hệ thống sẽ kiểm tra dữ liệu đầu vào theo nội dung sau:

|  |  |
| --- | --- |
| Tên trường | Yêu cầu kiểm tra |
| Nhóm thuốc | Nếu “Nhóm thuốc” để trống không chọn, hiển thị thông báo lỗi hình \* màu đỏ  Nếu “Nhóm thuốc” đã chọn, hiển thị thông tin nhóm thuốc đó. |
| Tên thuốc | Nếu “Tên thuốc” để trống, hiển thị thông báo lỗi hình \* màu đỏ  Nếu “Tên thuốc” nhập vào đã tồn tại trên hệ thống, hiển thị thông báo lỗi “Đã tồn tại” |
| Hoạt chất chính | Có thể nhập “Hoạt chất chính” phù hợp với tên thuốc |
| Nước sản xuất | Có thể nhập “Nước sản xuất” phù hợp với tên thuốc |
| Quy cách | Có thể nhập “Quy cách” phù hợp với tên thuốc |
| Hãng sản xuất | Có thể nhập “Hãng sản xuất” phù hợp với tên thuốc |
| Đơn giá | Nếu “Đơn giá” trống, hiển thị thông báo lỗi \* màu đỏ |
| Hàm lượng | Nếu “Hàm lượng” trống, hiển thị thông báo lỗi \* màu đỏ |
| Đường dùng | Nếu “Đường dùng” trống, hiển thị thông báo lỗi \* màu đỏ |

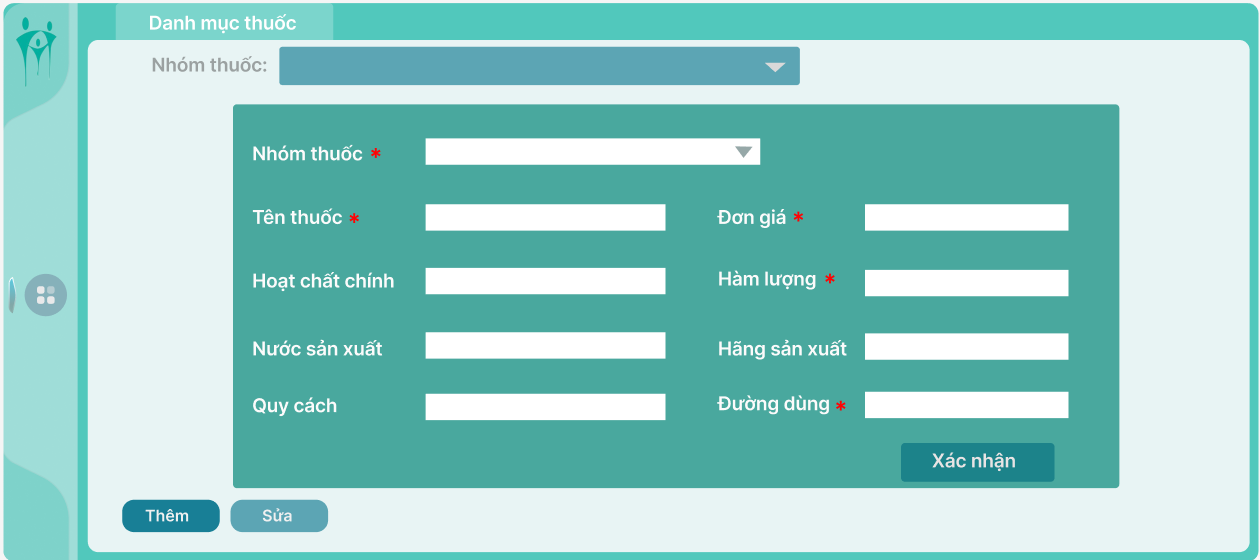
* + **Bước 4:** User click vào nút “Xác nhận”
  + **Bước 5:** Hệ thống lưu thông tin thuốc vào cơ sở dữ liệu và hiển thị tên thuốc đó trên danh mục thuốc.
* **Giao diện:**



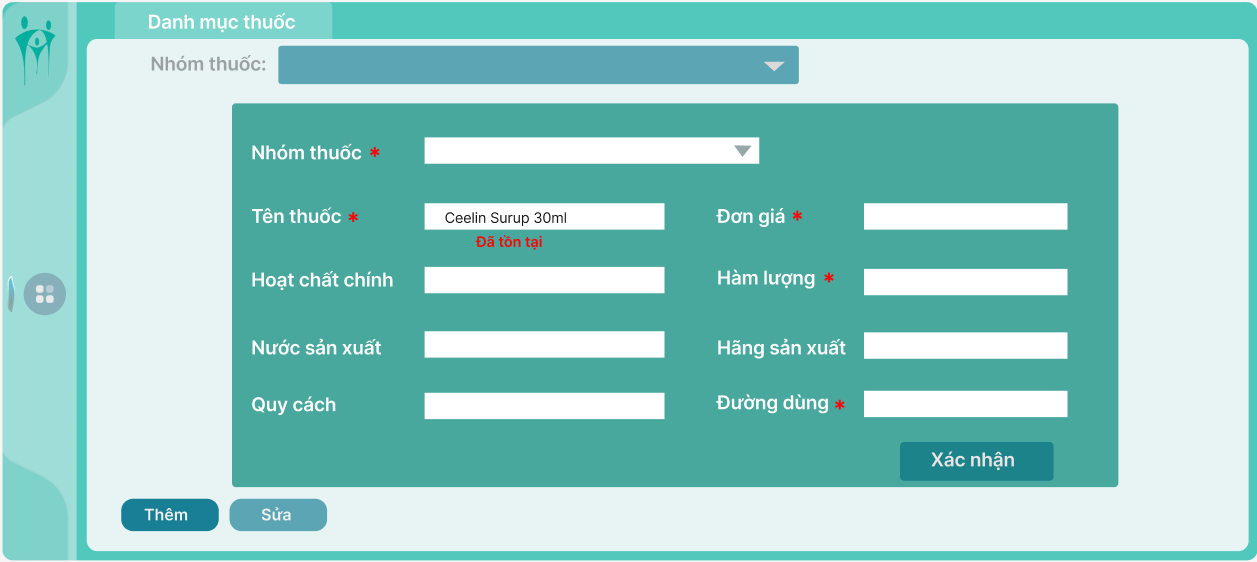
Hình 3.2. Màn hình giao diện khi ấn “Thêm”



Hình 3.3. Màn hình giao diện khi cập nhật các thông tin



Hình 3.4 Màn hình giao diện khi các trường bắt buộc không nhập thông tin



Hình 3.5. Màn hình giao diện khi các Tên thuốc đã tồn tại trên hệ thống

Ảnh có chứa bàn

Mô tả được tạo tự động

Hình 3.6. Hiển thị đã thêm tên thuốc ở màn hình danh mục thuốc

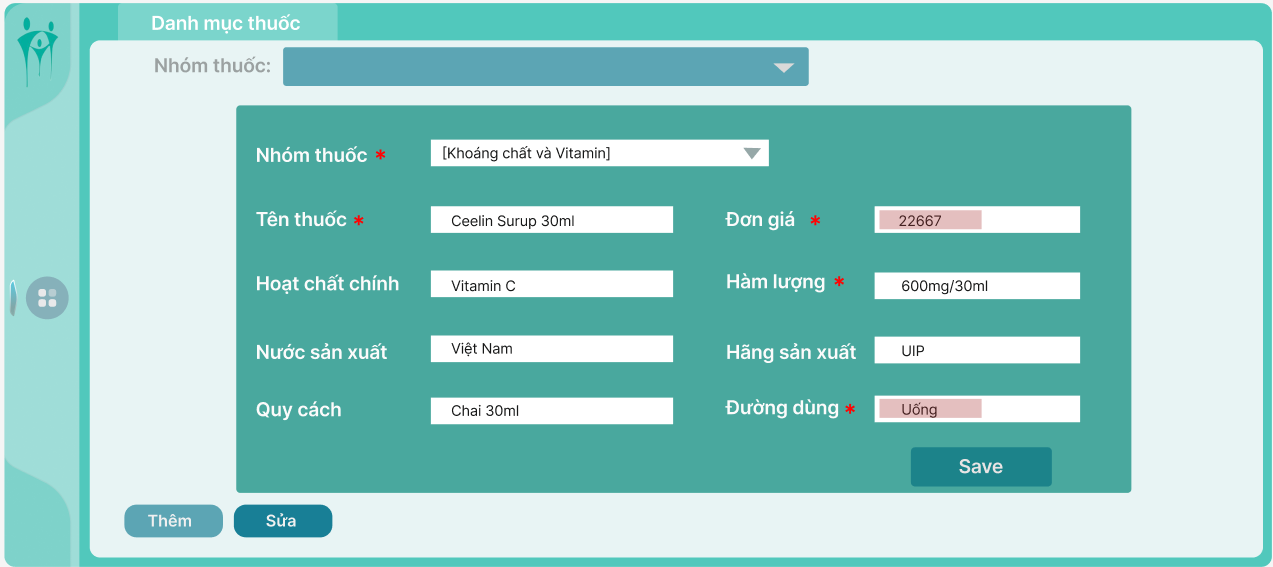
### Phân tích use case “Sửa”

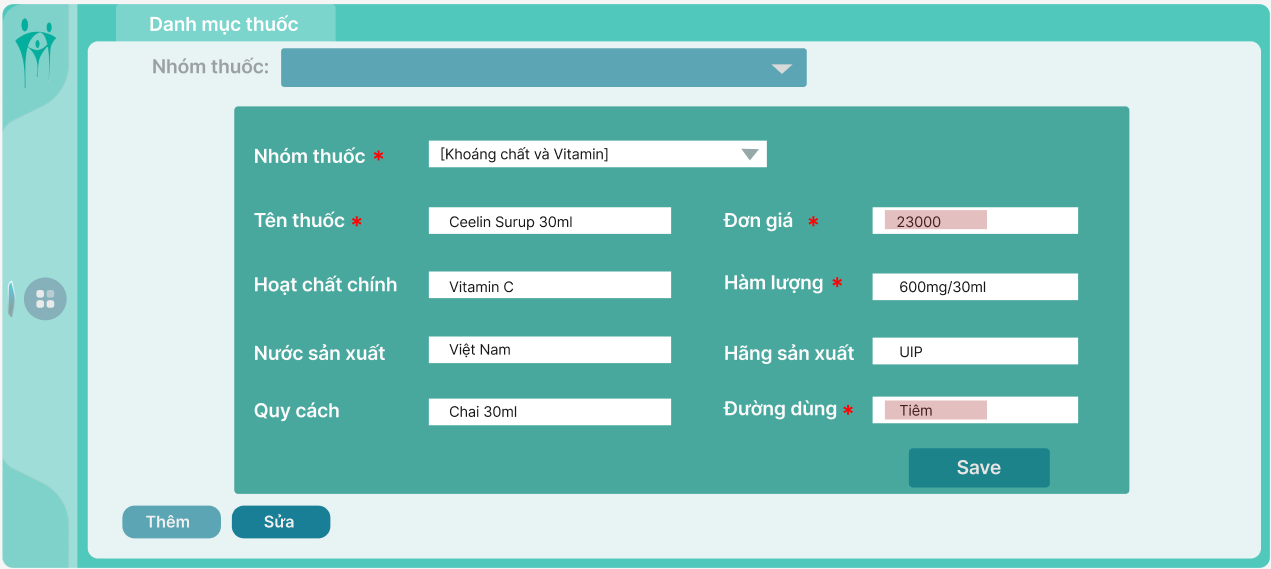
* **Tác nhân:** Nhân viên y tế
* **Điều kiện tiên quyết:** Đăng nhập vào hệ thống thành công
* **Mô tả khái quát:** Use case này sử dụng để nhân viên y tế có thể chỉnh sửa các loại thuốc trong danh mục thuốc bằng cách sửa một số chi tiết cần thiết như quy cách, hàm lượng dùng,….
* **Mô tả chi tiết:**
  + **Bước 1:** Nhấn đúp chuột vào loại thuốc mà cần sửa, sau đó thực hiện click vào button “Sửa” tại màn hình “Danh mục thuốc”

Ảnh có chứa bàn

Mô tả được tạo tự động

* + **Bước 2:** Hệ thống sẽ hiển thị màn hình, tại đây User sẽ chọn click vào mục nào cần sửa;

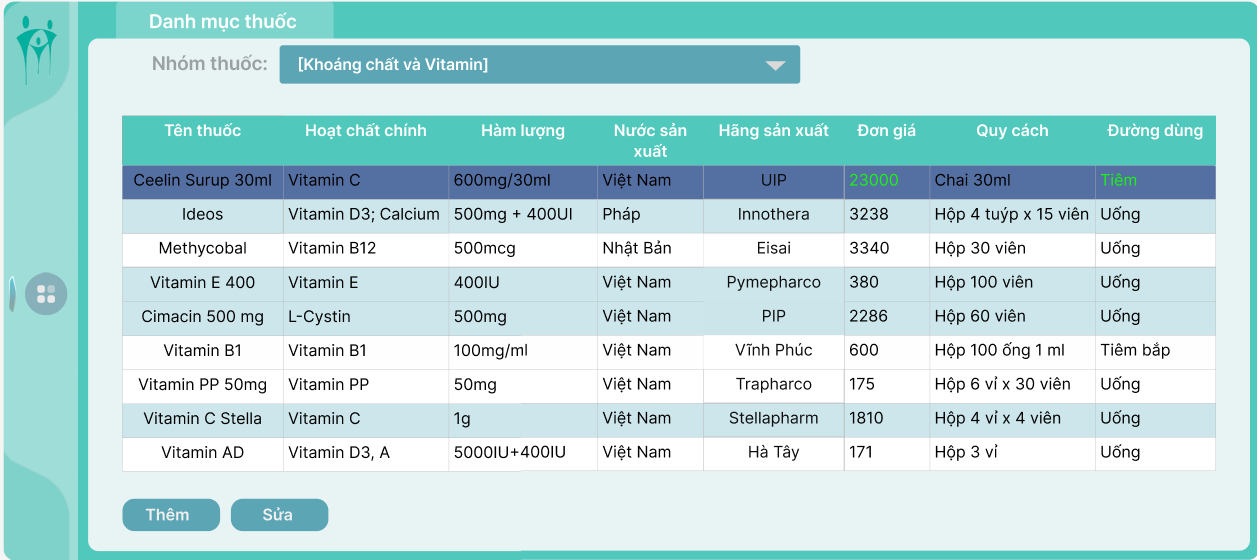
****

****

* + **Bước 3:** Hệ thống sẽ kiểm tra dữ liệu theo nội dung đã sửa như sau:

|  |  |
| --- | --- |
| Tên trường | Yêu cầu kiểm tra |
| Đơn giá | Đơn giá đã thay đổi so với lúc ban đầu hay chưa |
| Đường dùng | Đường dùng đã thay đổi so với lúc ban đầu từ “uống” chuyển sang “tiêm” |

* + **Bước 4:** User click vào nút “Save”
  + **Bước 5:** Hệ thống lưu thông tin thuốc đã thay đổi và hiển thị lại tên thuốc đó trên danh mục thuốc.
* **Giao diện;**

****

# THỰC HIỆN KIỂM THỬ HỆ THỐNG

## Lập kế hoạch kiểm thử

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Đề tài đã thực hiện được …

# TÀI LIỆU THAM KHẢO

* + - 1. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang
      2. Tên tác giả, Tên tác giả… *Tên công trình*. Nơi xuất bản, năm, trang

# PHỤ LỤC