



TRƯỜNG ĐẠI HỌC SÀI GÒN

Khoa Công nghệ thông tin

BÁO CÁO CÁ NHÂN

HỌC PHẦN: CƠ SỞ DỮ LIỆU NÂNG CAO

Chủ đề: QUARANTINE CAMP DATABASE

Giảng viên hướng dẫn: *Phan Trọng Nhân*

Sinh viên thực hiện: *Đỗ Phúc Thuận*

Nhóm: 6

MSSV: 3121411204

Lớp: *DCT121C3*

Khóa: 21

TP. Hồ Chí Minh, tháng 12/2023

MỤC LỤC

DANH MỤC HÌNH	6
DANH MỤC BẢNG	7
LỜI MỞ ĐẦU	8
I/ PHÂN TÍCH CHUNG	9
1) Assignment Specification	9
2) Assignment Requirements	10
2.1) Requirements chung	10
2.2) Requirements riêng cho đề tài Quarantine Camp Database	11
3) Tiến hành dịch thuật	11
4) Công việc được giao	12
5) Công nghệ hỗ trợ	13
6) Kết luận	13
II/ CƠ SỞ LÝ THUYẾT	13
1) Visual Studio Code	13
2) XAMPP	13
3) PHPMyAdmin	15
4) SQL	16
5) DBDiagram.io	16
III/ LƯỢC ĐỒ THỰC THỂ KẾT HỢP (ERD)	17
1) Các thực thể (Entities):	17
2) Phân tích các thực thể và thuộc tính	17
2.1) Thực thể BỆNH NHÂN	17
2.2) Thực thể PHÒNG	18
2.3) Thực thể XÉT NGHIỆM (thực thể yếu phụ thuộc Bệnh Nhân)	18
3) Mối quan hệ giữa các thực thể	19
3.1) Mối quan hệ giữa BỆNH NHÂN và XÉT NGHIỆM	19
3.2) Mối quan hệ giữa BỆNH NHÂN và các NHÂN VIÊN	20
3.2.1) BỆNH NHÂN và Nurse (Y tá)	20
3.2.2) BỆNH NHÂN và STAFF	21

3.2.3) BỆNH NHÂN và DOCTORS (Bác sĩ)	22
3.2.4) BỆNH NHÂN và PHÒNG	23
4) Lược đồ thực thể kết hợp ERD toàn bộ hệ thống.	24
IV/ THIẾT KẾ CƠ SỞ DỮ LIỆU VẬT LÝ	25
1) Chuyển đổi ERD sang RDM	25
2) Bảng xác định thuộc tính các thực thể	26
3) Bảng thể hiện Datatype của CSDL	27
3.1) Datatype của thực thể BỆNH NHÂN	27
3.2) Datatype của thực thể BỆNH LÝ ĐI KÈM	28
3.3) Datatype của thực thể TRIỆU CHỨNG THƯỜNG	28
3.4) Datatype của thực thể TRIỆU CHỨNG NGHIÊM TRỌNG	28
3.5) Datatype của thực thể XÉT NGHIỆM	28
3.6) Datatype của thực thể PHÒNG	29
3.7) Datatype của thực thể Staff_PhanVao_Phong	29
4) Demo data cho các tables.	29
4.2) Demo data cho table BenhLyDiKem	30
4.3) Demo data cho table TrieuChungThuong	30
4.4) Demo data cho table TrieuChungNghiemTrong	31
4.5) Demo data cho table XÉT NGHIỆM	31
4.6) Demo data cho table PHÒNG	31
4.7) Demo data cho table Staff_PhanVao_Phong	32
5) Relational Mapping	32
V/ SQL COMMANDS	34
1) Create Table	34
1.1) Create Table BENH NHÂN	34
1.2) Create Table BenhLyDiKem	35
1.3) Create Table TrieuChungThuong	35
1.4) Create Table TrieuChungNghiemTrong	36
1.5) Create Table XÉT NGHIỆM	36
1.6) Create Table PHÒNG	37

1.7) Create Table Staff_PhanVao_Phong	38
1.8) Create Table XetNghiem	38
1.9) Create Table Phong	39
1.10) Create Table NhanVien	40
1.11) Create Table staff, Doctors, Nurse, Manager, Volunteers	40
1.12) Create Table Thuoc	41
1.13) Create Table Dieutri	41
1.14) Create Table user	42
2) Insert	42
2.1) Insert into Table BENHNNHAN	42
2.2) Insert intoTable BenhLyDiKem	43
2.3) Insert intoTable TrieuChungThuong	43
2.4) Insert intoTable TrieuChungNghiemTrong	44
2.5) Insert intoTable XetNghiem	44
2.6) Insert intoTable Phong	45
2.7) Insert intoTable Staff_PhanVao_Phong	45
2.8) Insert into nhanvien	45
2.9) Insert into staff	47
2.10) Insert into nurse	47
2.12) Insert into manager	47
2.13) Insert into doctors	47
2.14) Insert into voulunteers	47
2.15) Insert into table phong	48
2.19) Insert into table thuoc	48
3) Alter Table	49
3.1) Alter Table BENHNNHAN	49
3.2) Alter Table XetNghiem	49
3.3) Alter Table BenhLyDiKem	49
3.4) Alter Table TrieuChungThuong	49
3.5) Alter Table TrieuChungNghiemTrong	50

3.6) Alter Table Staff_PhanVao_Phong	50
3.7) Alter Table staff, doctors, nurse, manager, volunteers	50
3.8) Alter Table Dieutri	51
VI/ TÀI LIỆU THAM KHẢO	51
VII/ KẾT LUẬN	51

DANH MỤC HÌNH

Hình 1: Xampp

Hình 2: PHPMyAdmin

Hình 3: DBDiagram.io

Hình 4: Thực thể BỆNH NHÂN

Hình 5: Thực thể PHÒNG

Hình 6: Thực thể XÉT NGHIỆM

Hình 7: Quan hệ giữa BN và XN

Hình 8: Quan hệ giữa BỆNH NHÂN và NURSE

Hình 9: Quan hệ giữa BỆNH NHÂN và STAFF

Hình 10: Quan hệ giữa BỆNH NHÂN, DOCTORS và THUỐC

Hình 11: Quan hệ giữa BỆNH NHÂN, PHÒNG và STAFF

Hình 12: ERD hoàn chỉnh

Hình 13: Relational Data Mapping

DANH MỤC BẢNG

Bảng 1: Bảng thuộc tính của thực thể và mối quan hệ

Bảng 2: Bảng kiểu dữ liệu của table BỆNH NHÂN

Bảng 3: Bảng kiểu dữ liệu của table BỆNH LÝ ĐI KÈM

Bảng 4: Bảng kiểu dữ liệu của table TRIỆU CHỨNG THƯỜNG

Bảng 5: Bảng kiểu dữ liệu của table TRIỆU CHỨNG NGHIÊM TRỌNG

Bảng 6: Bảng kiểu dữ liệu của table XÉT NGHIỆM

Bảng 7: Bảng kiểu dữ liệu của table PHÒNG

Bảng 8: Bảng kiểu dữ liệu của table Staff_PhanVao_Phong

Bảng 9: Bảng kiểu demo data của table BỆNH NHÂN

Bảng 10: Bảng kiểu demo data của table BenhLyDiKem

Bảng 11: Bảng kiểu demo data của table TrieuChungThuong

Bảng 12: Bảng kiểu demo data của table TrieuChungNghiemTrong

Bảng 13: Bảng kiểu demo data của table XetNghiem

Bảng 14: Bảng kiểu demo data của table PHÒNG

Bảng 15: Bảng kiểu demo data của table Staff_PhanVao_Phong

LỜI MỞ ĐẦU

Em tên là Đỗ Phúc Thuận, MSSV: 3121411204, Lớp: DCT121C3, khóa 21.

Trong bài báo cáo cá nhân này em xin trình bày những việc mình đã làm để góp phần hoàn thành đồ án lớn của nhóm.

Đồ án nhóm của em có chủ đề là “**Thiết kế hệ thống Quarantine Camp Database**”. Đây là một đề tài có tính thực tiễn cao, vì nước ta đã trải qua một mùa dịch bệnh với nhiều trại cách ly. Đề tài này giúp em cũng như cả nhóm tìm hiểu được nghiệp vụ của một khu cách ly và thiết kế được hệ thống cũng như database cho nó. Trong quá trình thực hiện đồ án, em đã cố gắng hết sức để hoàn thành tốt nhiệm vụ của mình, góp phần vào thành công chung của nhóm.

Bài báo cáo của em gồm 6 phần:

- + Phần I: PHÂN TÍCH CHUNG
- + Phần II: CƠ SỞ LÝ THUYẾT
- + Phần III: LƯỢC ĐỒ THỰC THỂ KẾT HỢP (ERD)
- + Phần IV: THIẾT KẾ CƠ SỞ DỮ LIỆU VẬT LÝ
- + Phần V: SQL COMMANDS
- + Phần VI: KẾT LUẬN

Em hy vọng rằng những việc em đã làm góp phần quan trọng vào thành công của đồ án. Qua quá trình làm đồ án em đã học hỏi được rất nhiều kinh nghiệm trong việc tìm hiểu nghiệp vụ, phân tích nghiệp vụ và thiết kế cơ sở dữ liệu hoàn chỉnh cho một hệ thống. Những kinh nghiệm cũng như kiến thức này sẽ giúp ích cho em trong tương lai.

I/ PHÂN TÍCH CHUNG

1) Assignment Specification

Em tiếp nhận đề tài và tiến hành đọc Assignment Specification, đề tài của em có specification như sau:

Due to the Covid-19 outbreak, a quarantine camp has been set up to isolate and monitor people under investigation for 21 days. Those people admitted to the quarantine camp are called “patient”. The camp stores patient information including unique number, full name, identity number, phone, gender, and address. In addition, it wants to record the patient comorbidities (e.g., cancer, chronic lung diseases, diabetes, heart conditions, immunocompromised state) because they will put a patient in a high-risk situation. In parallel, a patient needs to be tracked with his or her symptoms such as fever, dry cough, tiredness, aches and pains, sore throat, diarrhoea, conjunctivitis, headache, loss of taste or smell, a rash on skin, or discoloration of fingers or toes. Some of them may be serious like difficulty breathing or shortness of breath, chest pain or pressure, and loss of speech or movement. Unlike comorbidity, a patient symptom is different from time to time.

The camp has different types of people: managers, doctors, nurses, staffs, and volunteers. One doctor will be designated as the head of the camp. Each has its own responsibility. Besides, the camp has several buildings, each has many floors and rooms. Each room has a limited capacity. There are three types of room: normal room, emergency room, and recuperation room. When admitted by a staff, a patient is assigned into a room based on his or her current condition. Sometimes, a patient is moved from his or her room to the emergency room or the recuperation room. So, it is important to track a patient location history. The camp needs to know the admission date, from where the patient is moved to the camp, the staff information, and the testing information if any. A staff may admit many patients, and a patient is admitted by a staff.

The testing information includes those as described below:

- PCR test: the result is true (positive) or false (negative). In case it is positive, the camp wants to track the corresponding cycle threshold (ct) value.
- Quick test: the result is true (positive) or false (negative). In case it is positive, the camp wants to track the corresponding cycle threshold (ct) value.

- SPO2: which is the percent saturation of oxygen in the blood. The test measures blood oxygen levels, indicated by percentage (%).
- Respiratory rate: it is measured by how many breaths per minute.

A patient may have many testing during his or her stay. If the SPO2 is smaller than 96% and the respiratory rate is larger than 20 breaths per minute, the patient is marked “warning” and needs a healthcare action from the doctors. In case the patient has no clinical sign and the test is either negative or positive whose cycle threshold is larger than 30, he or she will be discharged from the camp. Neither of them, the patient will be tested for every 3 days by Quick test. It is important to track the discharge date for each patient.

A patient can receive treatment from at least one doctor. A doctor can treat many patients at the same time, or sometimes, he has no patients to treat. The camp needs the details of each treatment such as: treatment period (start date and end date), result, and medications. Each patient is taken care of by a nurse; a nurse can take care of many inpatients at the same time. The information of a medication is also stored in the database. This information consists of a unique code, name of the medication, effects, price, and expiration date.

2) Assignment Requirements

Với đề tài này thì thầy có những yêu cầu (*Requirements*) như sau:

2.1) Requirements chung

1. Design a fully labelled (E)ERD according to your business description. The diagram has to show appropriate entities (with key attributes underlined), relationships, cardinality ratios, and optional & mandatory membership classes (3 points).
2. Mapping your (E)ER diagram above to a relational database schema and identify all constraints not shown in your (E)ER diagram (1 point).
3. Build an application with the following requirements (6 points):
 - Programming environment: optional (desktop, web, or mobile application).
 - Programming language: optional.
 - Students need to prepare data and scripts for demonstration at the reporting session.

2.2) Requirements riêng cho đề tài Quarantine Camp Database

1. Search patient information: Search results include the name, phone number and information about his/her comorbidities (1 point).
2. Add information for a new patient (1 point).
3. List details of all testing which belong to a patient (1 point).
4. Make a report that provides full information about the patient including demographic information, comorbidities, symptoms, testing, and treatment (1 point).
5. Proving one use-case of indexing efficiency in your scenarios (1 point)
6. Solving one use-case of database security in your scenarios (1 point)

3) Tiến hành dịch thuật

Trong quá trình đọc và tìm hiểu specification, em có sử dụng các công cụ dịch thuật như: Google Translate, Google Bard, Cambridge Dictionary. Để hỗ trợ cho việc tìm hiểu nghiệp vụ được dễ dàng hơn.

Nghệp vụ của đề tài được dịch sang ngôn ngữ Tiếng Việt:

Do dịch Covid-19 bùng phát, một khu cách ly đã được thành lập để cách ly và theo dõi những người đang được điều tra trong 21 ngày. Những người được đưa vào khu cách ly được gọi là "bệnh nhân". Khu cách ly lưu trữ thông tin bệnh nhân bao gồm mã số duy nhất, họ tên đầy đủ, số CMND, số điện thoại, giới tính và địa chỉ. Ngoài ra, khu cách ly còn muốn ghi nhận các bệnh lý đi kèm của bệnh nhân (ví dụ: ung thư, bệnh phổi mãn tính, tiểu đường, bệnh tim, suy giảm miễn dịch) vì chúng sẽ khiến bệnh nhân có nguy cơ cao.

Song song đó, cần theo dõi bệnh nhân với các triệu chứng như sốt, ho khan, mệt mỏi, đau nhức, đau họng, tiêu chảy, viêm kết mạc, nhức đầu, mất vị giác hoặc khứu giác, phát ban trên da hoặc đổi màu ngón tay hoặc ngón chân. Một số triệu chứng có thể nghiêm trọng như khó thở hoặc thở ngắn, đau hoặc tức ngực, mất khả năng nói hoặc cử động. Không giống như bệnh lý đi kèm, triệu chứng của bệnh nhân có thể khác nhau tùy theo thời gian.

Khu cách ly có nhiều loại nhân viên khác nhau: quản lý, bác sĩ, y tá, nhân viên và tình nguyện viên. Một bác sĩ sẽ được chỉ định làm trưởng khu cách ly. Mỗi người có trách nhiệm riêng của mình. Ngoài ra, khu cách ly có một số tòa nhà, mỗi tòa nhà có nhiều tầng và phòng. Mỗi phòng có sức chứa hạn chế. Có ba loại phòng: phòng thông thường, phòng cấp cứu và phòng hồi sức. Khi được nhân viên (Staff)

tiếp nhận, bệnh nhân sẽ được phân vào phòng dựa trên tình trạng hiện tại của mình. Đôi khi, bệnh nhân được chuyển từ phòng thông thường sang phòng cấp cứu hoặc phòng hồi sức. Vì vậy, việc theo dõi lịch sử vị trí của bệnh nhân là rất quan trọng.

Khu cách ly cần biết ngày nhập viện, bệnh nhân được chuyển đến khu cách ly từ đâu, thông tin nhân viên và thông tin xét nghiệm nếu có. Một nhân viên (Staff) có thể tiếp nhận nhiều bệnh nhân và một bệnh nhân được tiếp nhận bởi một nhân viên (Staff). Thông tin xét nghiệm bao gồm:

- + Xét nghiệm PCR: kết quả là dương tính (true) hoặc âm tính (false). Trong trường hợp dương tính, khu cách ly muốn theo dõi giá trị ngưỡng chu kỳ (ct) tương ứng.
- + Xét nghiệm nhanh: kết quả là dương tính (true) hoặc âm tính (false). Trong trường hợp dương tính, khu cách ly muốn theo dõi giá trị ngưỡng chu kỳ (ct) tương ứng.
- + SPO2: là độ bão hòa oxy trong máu. Xét nghiệm đo nồng độ oxy trong máu, được thể hiện bằng tỷ lệ phần trăm (%).
- + Tốc độ hô hấp: được đo bằng số lần thở mỗi phút.

Một bệnh nhân có thể có nhiều xét nghiệm trong thời gian lưu trú. Nếu SPO2 nhỏ hơn 96% và tốc độ hô hấp lớn hơn 20 nhịp thở mỗi phút, bệnh nhân được đánh dấu "cảnh báo" (?) và cần được bác sĩ chăm sóc y tế. Trong trường hợp bệnh nhân không có dấu hiệu lâm sàng và xét nghiệm âm tính hoặc dương tính với giá trị ngưỡng chu kỳ lớn hơn 30, bệnh nhân sẽ được xuất viện. Nếu không có dấu hiệu lâm sàng, bệnh nhân sẽ được xét nghiệm nhanh 3 ngày một lần. Việc theo dõi ngày xuất viện cho mỗi bệnh nhân là rất quan trọng.

Một bệnh nhân có thể được điều trị bởi ít nhất một bác sĩ. Một bác sĩ có thể điều trị cho nhiều bệnh nhân cùng lúc, hoặc đôi khi, không có bệnh nhân nào để điều trị. Khu cách ly cần biết chi tiết của từng lần điều trị như: thời gian điều trị (ngày bắt đầu và ngày kết thúc), kết quả và thuốc. Mỗi bệnh nhân được chăm sóc bởi một y tá; một y tá có thể chăm sóc nhiều bệnh nhân nội trú cùng lúc. Thông tin về thuốc cũng được lưu trữ trong cơ sở dữ liệu. Thông tin này bao gồm một mã duy nhất, tên thuốc, tác dụng, giá tiền và hạn sử dụng.

4) Công việc được giao

Em được phân công thiết kế ERD, thiết kế CSDL vật lý, tạo database để import vào DBMS. Ngoài ra em cũng có hỗ trợ các bạn trong team trong công việc riêng của các bạn khác trong nhóm. Tài liệu tham khảo ở nhiều nguồn sẽ để ở cuối bài báo cáo.

5) Công nghệ hỗ trợ

Đề tài được cài đặt trên nền tảng web với XAMPP. Hệ quản trị CSDL là PHPMyAdmin. Ngôn ngữ Front-end chính là HTML/CSS/JS, back-end là PHP.

Ngoài ra những công nghệ khác hỗ trợ trong việc build đồ án như: Visual Studio Code, DBdiagram.io,....

6) Kết luận

Sau khi dịch thuật và đọc assignment specification thì em đã nắm được sơ nghiệp vụ của một Quarantine Camp. Sau đó em gửi bản dịch thuật lên cho các bạn khác trong nhóm đọc và phân tích. Bản dịch thuật của em đã có chọn lọc những từ ngữ sao cho phù hợp với ngữ cảnh, ngữ nghĩa nên rất dễ hiểu. Trong bản dịch thuật cũng có highlight những phần quan trọng để làm database. Phần tiếp theo em sẽ phân tích specification dựa trên bản dịch thuật.

II/ CƠ SỞ LÝ THUYẾT

1) Visual Studio Code

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn mở miễn phí được phát triển bởi Microsoft. Nó được phát hành vào năm 2015 và đã nhanh chóng trở thành một trong những trình soạn thảo mã nguồn phổ biến nhất.

VS Code hỗ trợ nhiều ngôn ngữ lập trình, bao gồm **JavaScript, Python, C/C++, Java, TypeScript, HTML, CSS**, và nhiều ngôn ngữ khác. Nó cũng có thể được sử dụng để chỉnh sửa các tệp mã nguồn cho các ứng dụng web, ứng dụng di động, ứng dụng máy tính để bàn, và nhiều loại ứng dụng khác.

VS Code có một số tính năng nổi bật, bao gồm:

- + Hỗ trợ nhiều ngôn ngữ lập trình
- + Tự hoàn thành code thông minh
- + Chỉnh sửa syntax
- + Sửa lỗi code
- + Extensions

VS Code là một công cụ mạnh mẽ và linh hoạt có thể được sử dụng cho nhiều mục đích khác nhau. Nó là một lựa chọn tuyệt vời cho các lập trình viên ở mọi cấp độ kinh nghiệm.

2) XAMPP

XAMPP là viết tắt của Cross-Platform, Apache, MySQL, PHP. Nó là một phần mềm mã nguồn mở và miễn phí cho phép chạy một môi trường web cục bộ trên máy tính cá nhân mà không cần phải kết nối với internet hoặc đăng ký dịch vụ lưu trữ web.

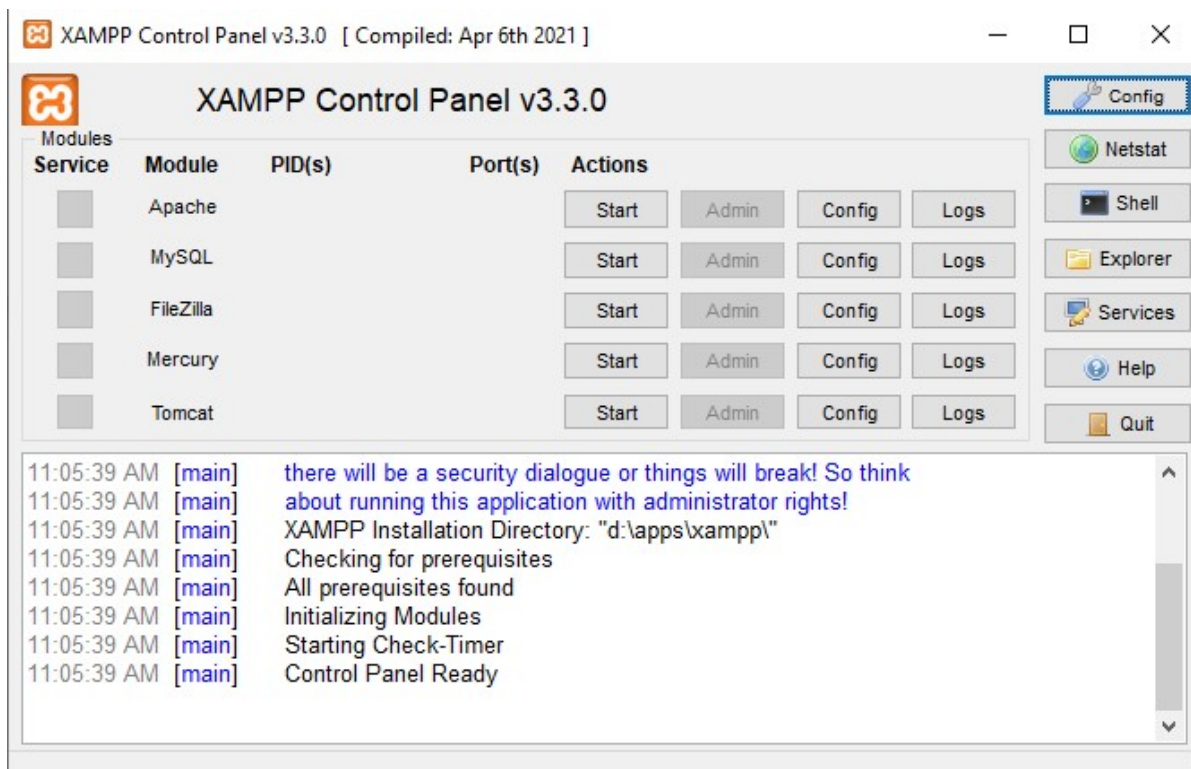
XAMPP bao gồm các thành phần sau:

- + Apache: Một máy chủ web phổ biến.
- + MySQL: Một cơ sở dữ liệu quan hệ.
- + PHP: Ngôn ngữ lập trình.

Để cài đặt XAMPP, hãy truy cập trang web của XAMPP (<https://www.apachefriends.org/download.html>) và tải xuống phiên bản phù hợp với hệ điều hành. Sau khi tải xuống, hãy chạy tệp cài đặt và làm theo hướng dẫn trên màn hình.

Sau khi cài đặt, bật lên và bấm vào Explorer, tại đây tìm thư mục “htdocs” nếu chưa có thì tạo. Thư mục này sẽ là thư mục gốc cho các trang web và ứng dụng web. Để truy cập trang web, hãy mở trình duyệt web và nhập địa chỉ sau: `[localhost]` sẽ thấy trang web mặc định của XAMPP.

Để truy cập vào web của mình thì `[localhost/path-to-your-web-folder]`



Hình 1: XAMPP

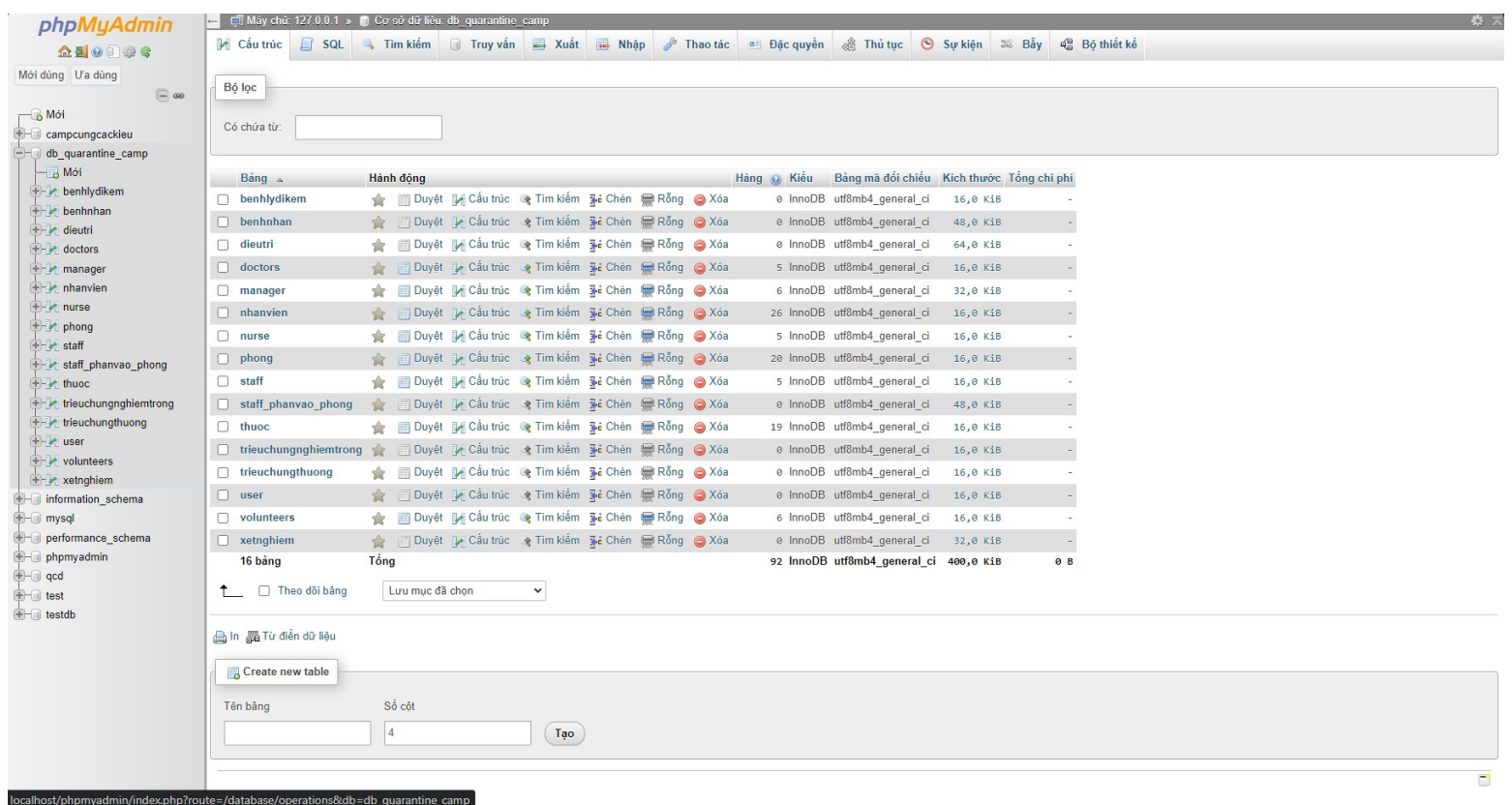
3) PHPMYAdmin

PHPMYAdmin là một công cụ mã nguồn mở miễn phí được viết bằng ngôn ngữ lập trình PHP để xử lý các tác vụ liên quan đến quản trị CSDL của MySQL thông qua trình duyệt web. Được tích hợp sẵn trong XAMPP. Là công cụ mạnh mẽ và dễ sử dụng cho phép tạo, sửa đổi, xóa và quản lý các bảng, hàng và cột trong cơ sở dữ liệu.

PHPMYAdmin có một giao diện web trực quan dễ dàng sử dụng. PHPMYAdmin cung cấp những tính năng sau:

- + Tạo cơ sở dữ liệu mới
- + Import và export cơ sở dữ liệu
- + Quản lý người dùng cơ sở dữ liệu
- + Quản lý các bảng và cột trong cơ sở dữ liệu
- + Thực thi các truy vấn SQL

Để truy cập vào PHPMYAdmin thì ghi vào trình duyệt địa chỉ sau:
[<http://localhost/phpmyadmin/>]



Hình 2: PHPMYAdmin

4) SQL

SQL là viết tắt của Structured Query Language, là một ngôn ngữ lập trình được sử dụng để giao tiếp với cơ sở dữ liệu quan hệ. Nó được sử dụng để tạo, truy vấn, cập nhật và xóa dữ liệu trong cơ sở dữ liệu.

Các câu lệnh SQL cơ bản bao gồm:

SELECT: Dùng để truy vấn dữ liệu từ cơ sở dữ liệu

INSERT: Dùng để thêm dữ liệu vào cơ sở dữ liệu

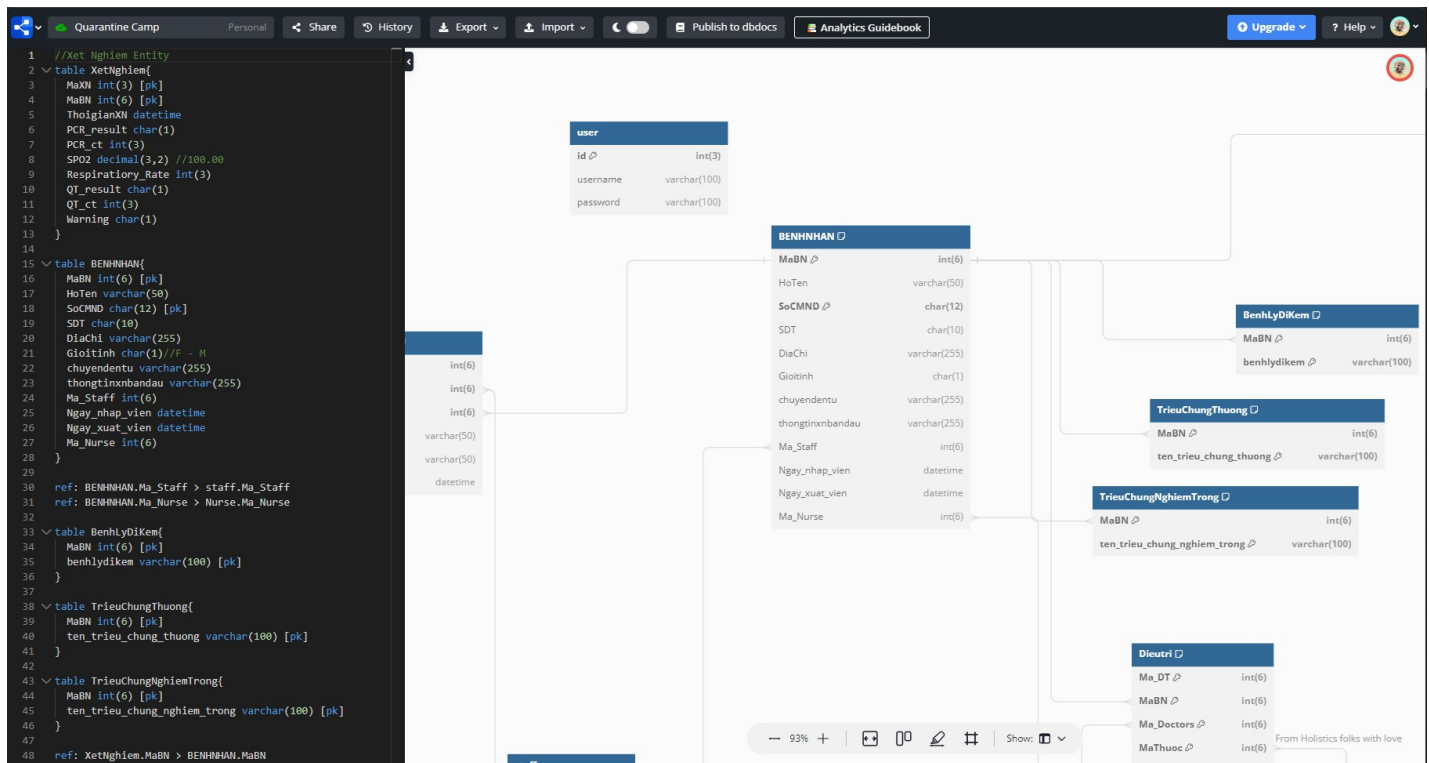
UPDATE: Dùng để cập nhật dữ liệu trong cơ sở dữ liệu

DELETE: Dùng để xóa dữ liệu khỏi cơ sở dữ liệu

5) DBDiagram.io

<https://dbdiagram.io/> là một trang web cung cấp công cụ trực tuyến miễn phí để thiết kế sơ đồ cơ sở dữ liệu. Công cụ này cho phép tạo sơ đồ CSDL quan hệ bằng cách sử dụng một ngôn ngữ mô hình hóa domain-specific (DSL).

DSL của dbdiagram.io rất dễ học và sử dụng. Có thể bắt đầu bằng cách tạo một bảng mới và thêm các cột, khóa chính và khóa ngoại. Cũng có thể thêm các mối quan hệ giữa các bảng.



Hình 3: DBDiagram.io

Sau khi hoàn thành sơ đồ, có thể xuất nó sang định dạng tệp khác nhau, chẳng hạn như PNG, PDF hoặc MySQL. Cũng có thể chia sẻ sơ đồ của mình với những người khác bằng cách cung cấp cho họ URL của nó.

III/ LƯỢC ĐỒ THỰC THỂ KẾT HỢP (ERD)

1) Các thực thể (Entities):

Sau khi phân tích yêu cầu đề bài thì em có thành lập được những thực thể sau:

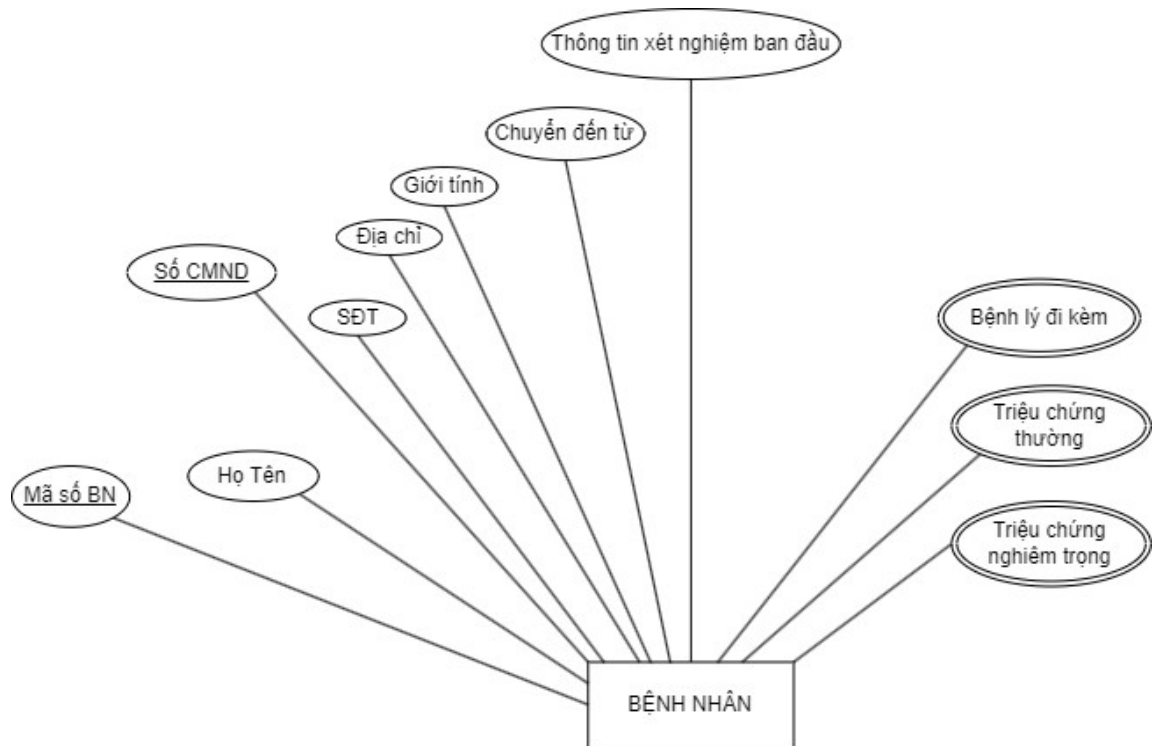
- **Bệnh nhân:** Lưu trữ thông tin của bệnh nhân được cách ly, bao gồm mã bệnh nhân, họ tên, số CMND, số điện thoại, giới tính, địa chỉ, các bệnh lý đi kèm, triệu chứng thường, triệu chứng nghiêm trọng.
- **Nhân viên:** Lưu trữ thông tin của nhân viên khu cách ly, bao gồm mã nhân viên, họ tên, chức vụ, ngày bắt đầu làm việc, địa chỉ, giới tính, số điện thoại. Trong people có các specialization disjoint khác như Staff, Doctors, Managers, Volunteer và Nurse.
- **Phòng:** Lưu trữ thông tin của các phòng trong khu cách ly, bao gồm mã phòng, loại phòng, sức chứa, tòa nhà, tầng.
- **Xét nghiệm:** Là thực thể yếu phụ thuộc vào Bệnh Nhân, lưu trữ thông tin của các lần xét nghiệm của bệnh nhân, bao gồm: PCR result, PCR ct, QT result, QT ct (Quick Test), respiratory rate, SPO2 và Cảnh báo
- **Thuốc:** Lưu trữ thông tin của các loại thuốc được sử dụng trong điều trị, bao gồm mã thuốc, tên thuốc, tác dụng, giá tiền, hạn sử dụng.

Sau khi xác định được các thực thể thì em chuyển thông tin này lên nhóm để phân chia công việc. Em được nhóm trưởng phân cho việc phân tích 3 thực thể: **Bệnh nhân, Phòng và Xét nghiệm**

2) Phân tích các thực thể và thuộc tính

2.1) Thực thể BỆNH NHÂN

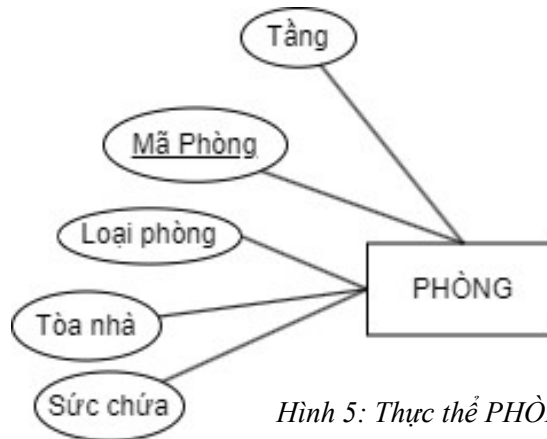
Thực thể BỆNH NHÂN gồm các thuộc tính sau: Mã số BN, Số CMND, Họ Tên, SĐT, địa chỉ, giới tính, chuyên đến từ, thông tin XN ban đầu, bệnh lý đi kèm, triệu chứng thường, triệu chứng nghiêm trọng.



Hình 4: Thực thể BỆNH NHÂN

2.2) Thực thể PHÒNG

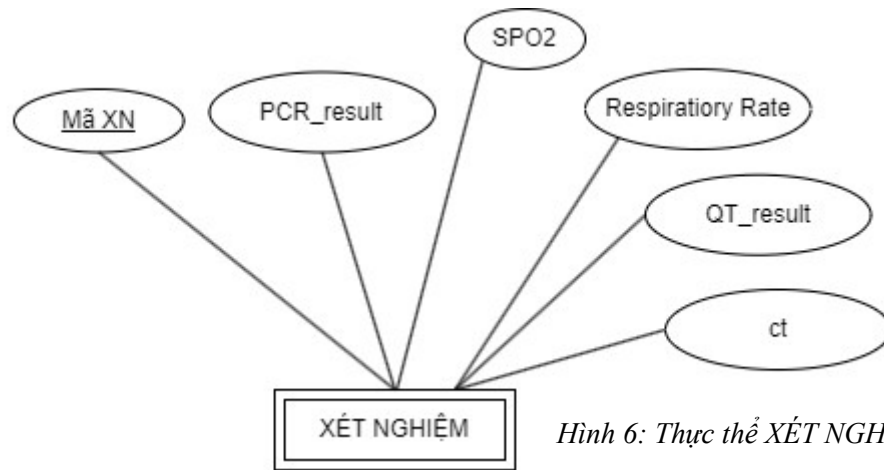
Thực thể PHÒNG gồm các thuộc tính sau: Mã Phòng, tầng, loại phòng, tòa nhà, sức chứa.



Hình 5: Thực thể PHÒNG

2.3) Thực thể XÉT NGHIỆM (thực thể yếu phụ thuộc Bệnh Nhân)

Thực thể XÉT NGHIỆM gồm các thuộc tính sau: PCR_result, PCR_ct, SPO2, Respiratory Rate, QT_result, QT_ct

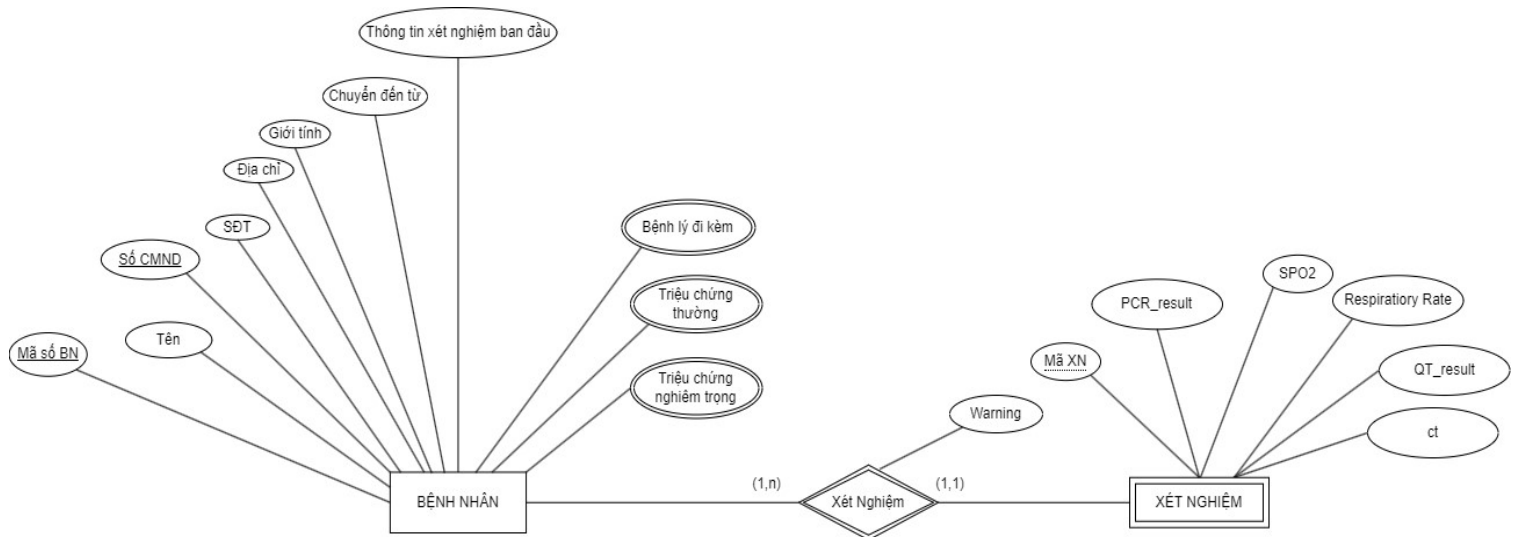


Hình 6: Thực thể XÉT NGHIỆM

3) Mối quan hệ giữa các thực thể

3.1) Mối quan hệ giữa BỆNH NHÂN và XÉT NGHIỆM

BỆNH NHÂN thực hiện XÉT NGHIỆM. BỆNH NHÂN có thể thực hiện một hoặc nhiều XÉT NGHIỆM nhưng 1 record XÉT NGHIỆM chỉ thuộc về duy nhất một BỆNH NHÂN. Do XÉT NGHIỆM là thực thể yếu phụ thuộc vào BỆNH NHÂN nên được vẽ nét đôi, và lấy khóa của thực thể BỆNH NHÂN làm khóa chính. Trong quá trình thực thi xét nghiệm, căn cứ vào các chỉ số, bệnh nhân sẽ có warning khác nhau.

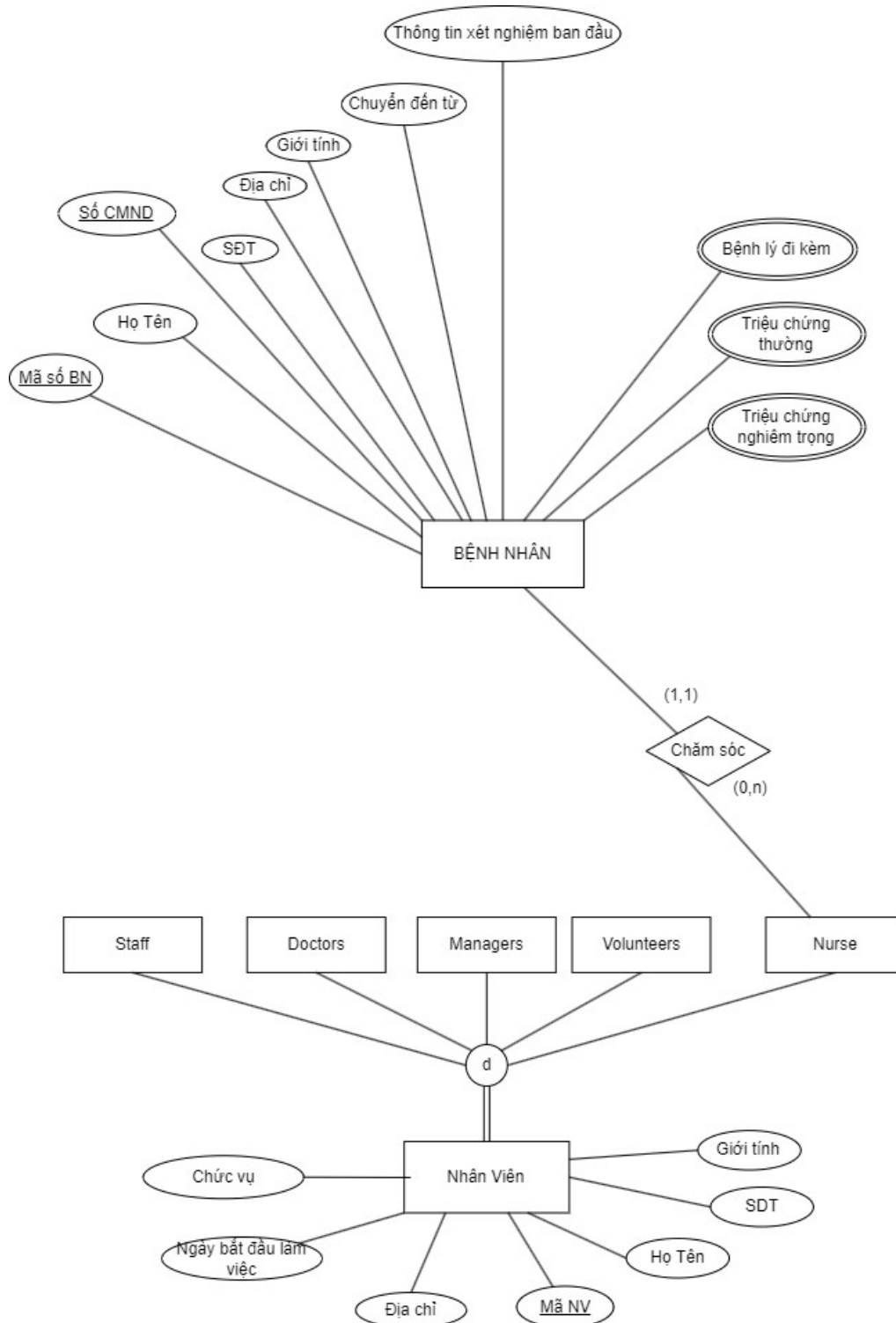


Hình 7: Quan hệ giữa BN và XN

3.2) Môi quan hệ giữa BỆNH NHÂN và các NHÂN VIÊN

3.2.1) BỆNH NHÂN và Nurse (Y tá)

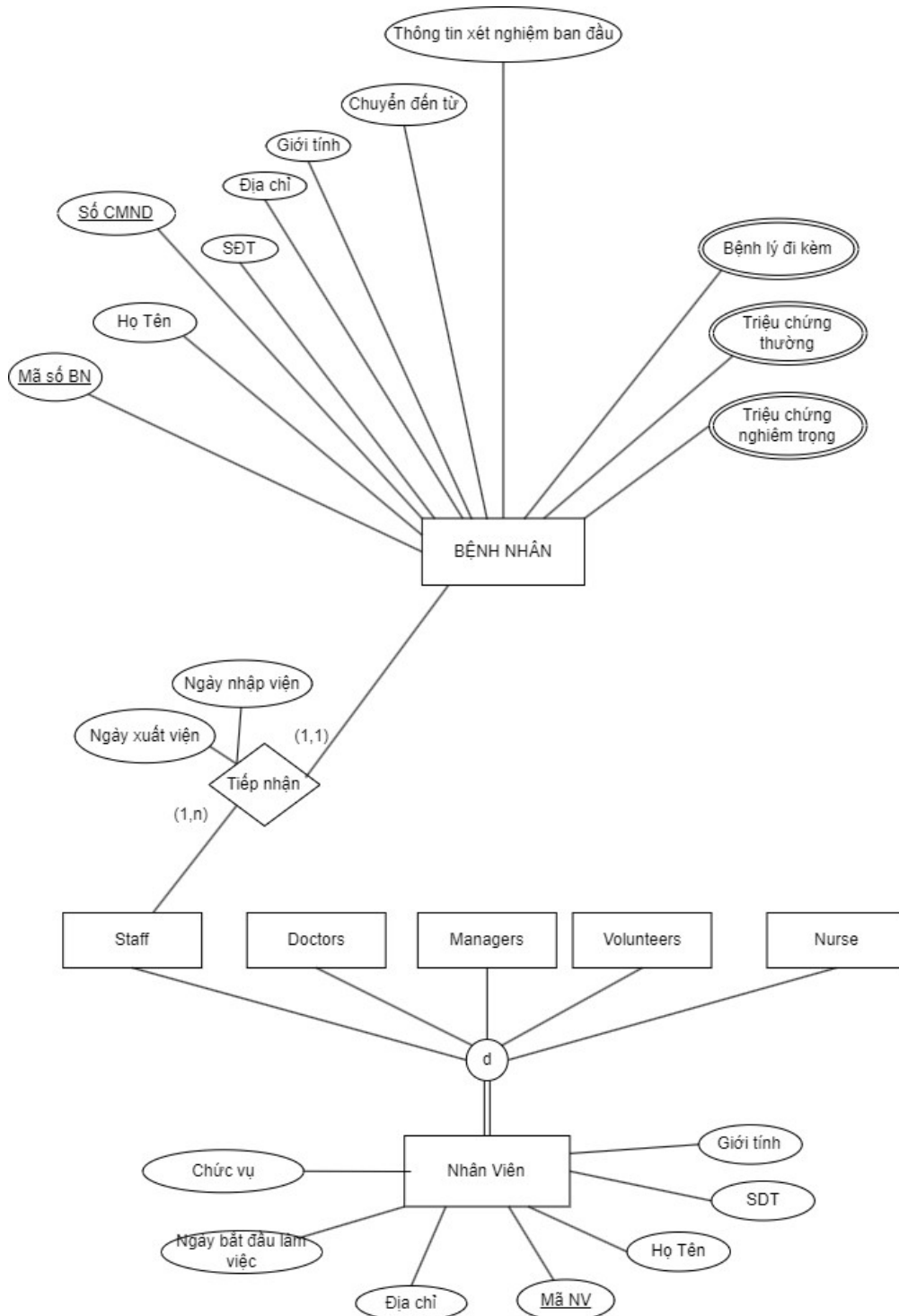
Nurse chăm sóc bệnh nhân. Một Nurse có thể chăm sóc 1 hoặc nhiều bệnh nhân cùng lúc, một bệnh nhân được chăm sóc bởi một nurse.



Hình 8: Quan hệ giữa BỆNH NHÂN và NURSE

3.2.2) BỆNH NHÂN và STAFF

Một Staff có thể tiếp nhận một hoặc nhiều Bệnh Nhân, Một bệnh nhân được tiếp nhận bởi một Staff. Khi tiếp nhận ta cần biết ngày nhập viện và ngày xuất viện của bệnh nhân.

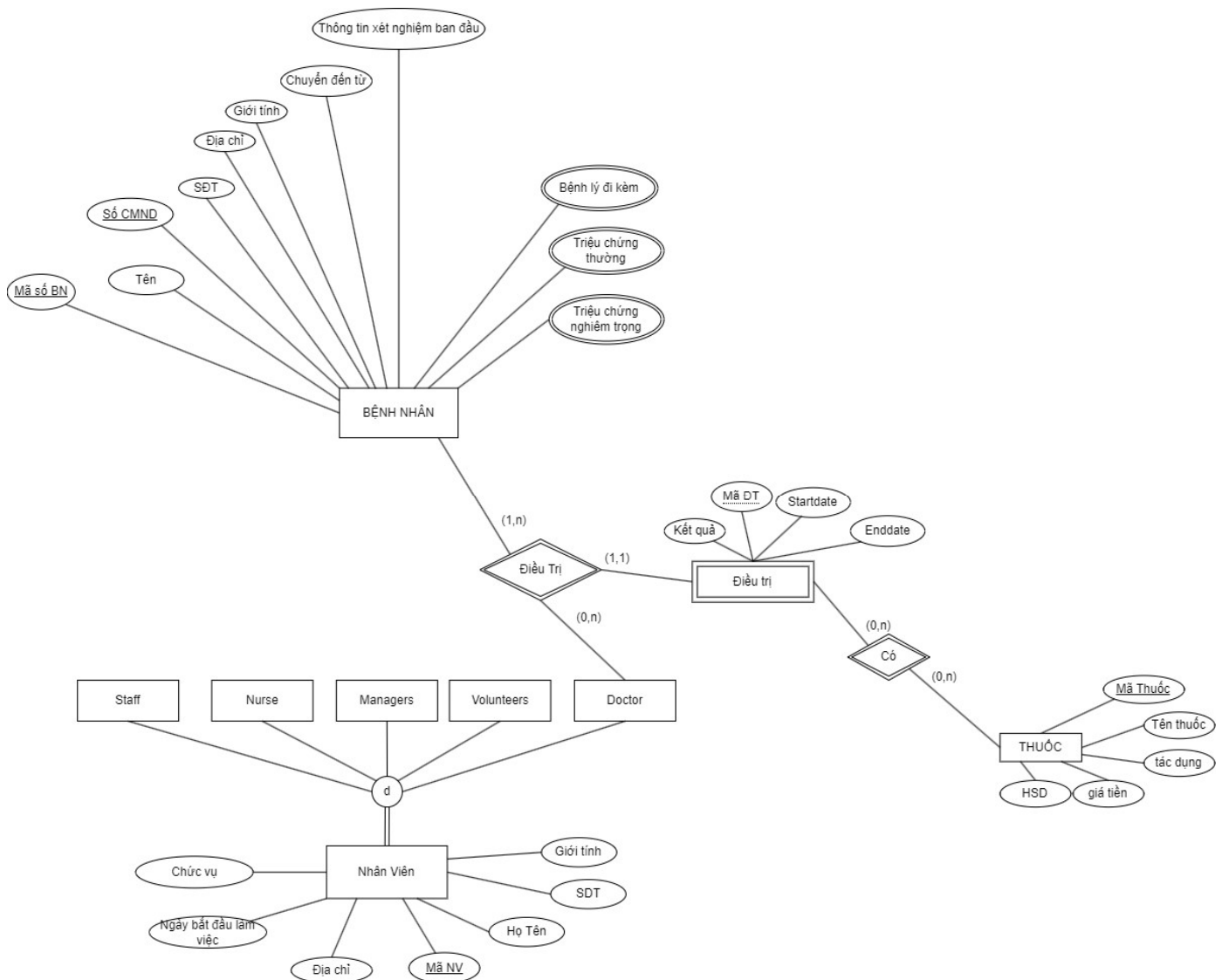


Hình 9: Quan hệ giữa BỆNH NHÂN và STAFF

3.2.3) BỆNH NHÂN và DOCTORS (Bác sĩ)

Một bệnh nhân có thể được điều trị bởi ít nhất một bác sĩ. Một bác sĩ có thể điều trị cho nhiều bệnh nhân cùng lúc, hoặc đôi khi, không có bệnh nhân nào để điều trị. Khu cách ly cần biết chi tiết của từng lần điều trị như: thời gian điều trị (Startdate và Enddate), kết quả và thuốc dùng trong lần điều trị đó.

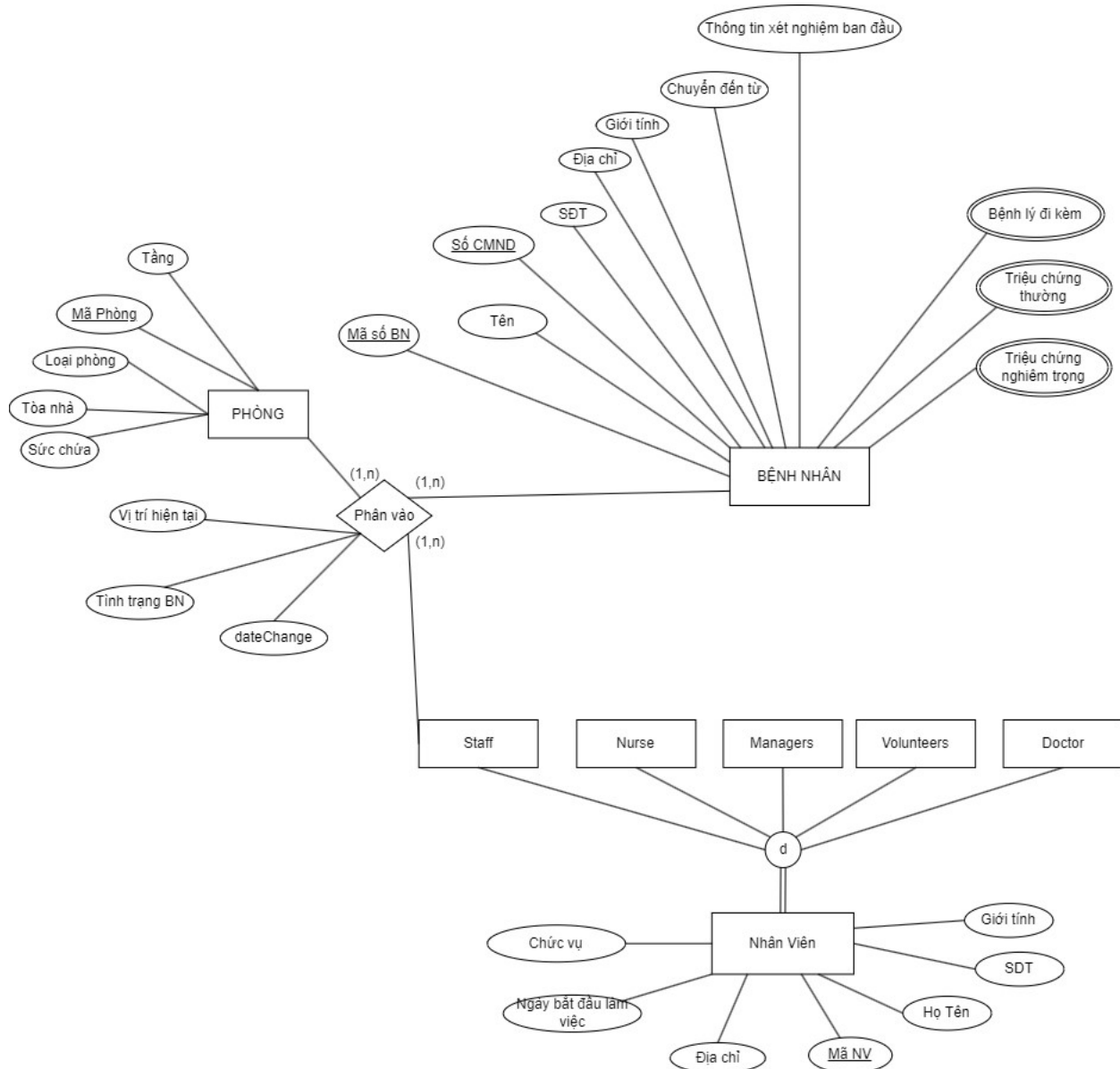
Do khu cách ly cần biết chi tiết của lần điều trị nên sinh ra thực thể yếu ĐIỀU TRỊ tham gia vào mỗi quan hệ ĐIỀU TRỊ phụ thuộc vào Bệnh Nhân và Doctors. Khi có điều trị thì mối quan hệ giữa THUỐC và ĐIỀU TRỊ mới thành lập, dẫn đến ĐIỀU TRỊ là thực thể yếu.



Hình 10: Quan hệ giữa BỆNH NHÂN, DOCTORS và THUỐC

3.2.4) BỆNH NHÂN và PHÒNG

Khi được nhân viên (Staff) tiếp nhận, BỆNH NHÂN sẽ được phân vào PHÒNG dựa trên tình trạng hiện tại. Đôi khi, BỆNH NHÂN được chuyển từ phòng thông thường sang phòng cấp cứu hoặc phòng hồi sức. Vì vậy, để theo dõi lịch sử vị trí của BỆNH NHÂN thì ở mỗi quan hệ Phân Vào sẽ có thuộc tính Vị trí hiện tại. Lưu trữ vị trí hiện tại của BỆNH NHÂN, khi cần thì truy vấn ra sẽ ra lịch sử di chuyển PHÒNG của BỆNH NHÂN



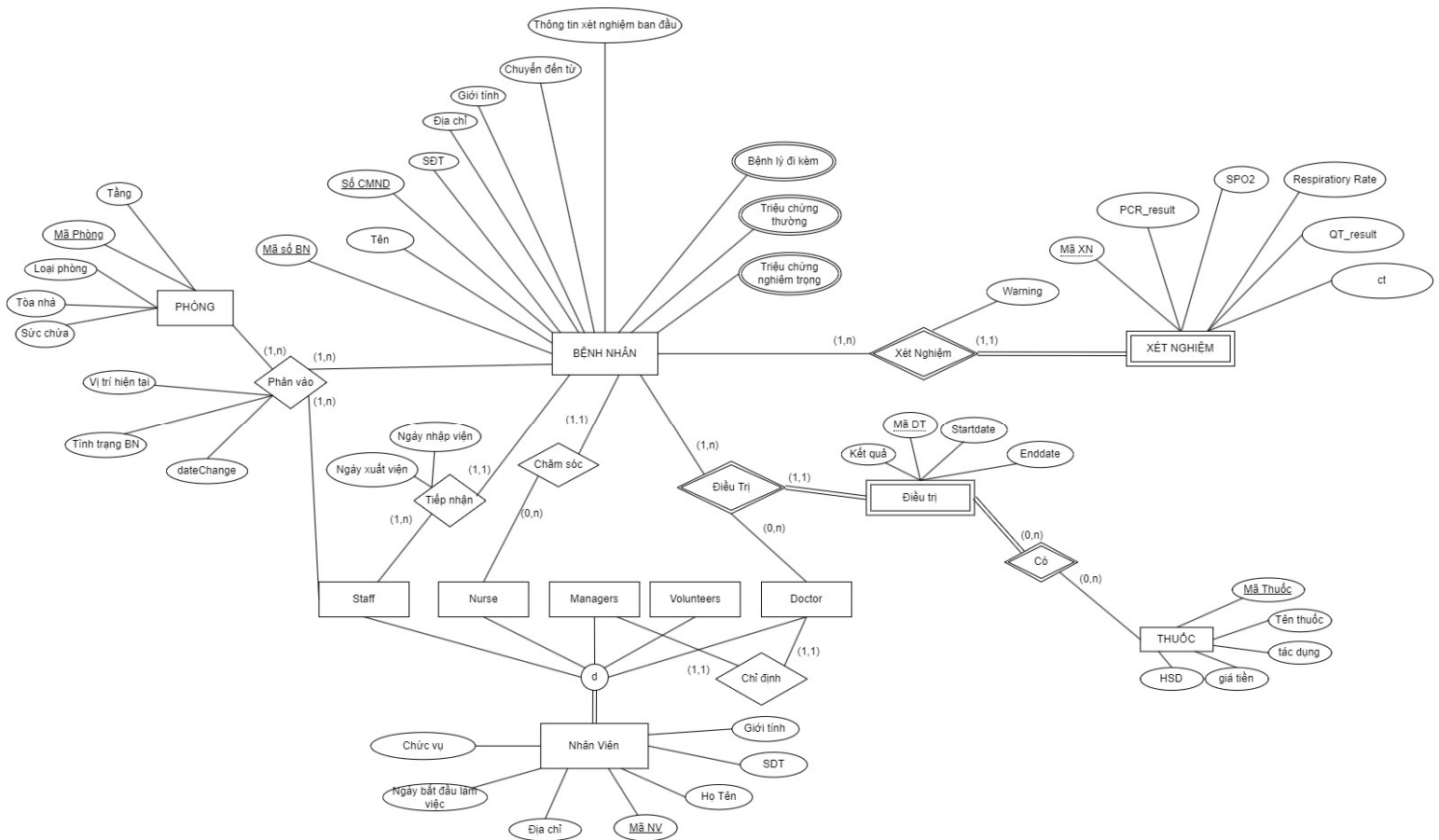
Hình 11: Quan hệ giữa BỆNH NHÂN, PHÒNG và STAFF

Thực thể PHÒNG sẽ tham gia vào mối quan hệ này, dẫn đến mối quan hệ này là một mối quan hệ bậc 3 với sự tham gia của 3 thực thể BỆNH NHÂN, PHÒNG và STAFF

4) Lược đồ thực thể kết hợp ERD toàn bộ hệ thống.

Sau khi hoàn thành phân tích 3 thực thể được giao thì em và team hợp lại để kết hợp thành một ERD hoàn chỉnh của hệ thống Quarantine Camp Database.

Dưới đây là lược đồ ERD hoàn chỉnh của toàn bộ hệ thống:



Hình 12: ERD hoàn chỉnh

IV/ THIẾT KẾ CƠ SỞ DỮ LIỆU VẬT LÝ

1) Chuyển đổi ERD sang RDM

Chuyển 3 thực thể: BỆNH NHÂN, XÉT NGHIỆM, PHÒNG và các mối liên hệ liên quan từ ERD sang RDM.

Thực thể Bệnh Nhân gồm những thuộc tính sẵn có và 2 mối quan hệ 1-n là Staff tiếp nhận và Nurse chăm sóc nên đem khóa của 2 thực thể Staff và Nurse sang làm khóa ngoại.

BỆNH NHÂN => BENHNNHAN(MaBN, SoCMND, Hoten, SDT, Diachi, Gioitinh, Chuyendentu, Thongtinxnbandau, *MaNurse*, *MaStaff*, Ngaynhapvien, Ngayxuatvien)

Trong thực thể Bệnh nhân có các thuộc tính đa trị nên được tách ra làm bảng mới.

Bệnh lý đi kèm => BENHLYDIKEM(MaBN, benhlydikem)

Triệu chứng thường => TRIEUCHUNGTHUONG(MaBN, trieuchungthuong)

Triệu chứng nghiêm trọng => TRIEUCHUNGNGHIEMTRONG(MaBN, trieuchungngkiemtrong)

Thực thể Xét Nghiệm gồm những thuộc tính sẵn có, do là thực thể yếu nên lấy khóa chính của thực thể mà nó phụ thuộc để làm khóa chính.

XÉT NGHIỆM => XETNGHIEM(MaXN, MaBN, ThoigianXN, PCR_result, SPO2, Respiratory Rate, QT_result, ct, Warning)

Thực thể Phòng gồm những thuộc tính sẵn có.

PHÒNG => PHONG(MaPhong, tang, loaiphong, toanha, succhua)

Mối quan hệ Staff_PhanVao_Phong là mối quan hệ bậc 3 với n-n-n nên ta tạo bảng mới, đem khóa của 3 thực thể tham gia vào làm khóa chính cho bảng của mối quan hệ.

Staff phân vào Phòng => Staff_PhanVao_Phong(MaPhong, Ma_Staff, MaBN, Tinhtranghientai, VittriBN, dateChange)

2) Bảng xác định thuộc tính các thực thể

Thực thể	Các thuộc tính
1. BENH NHAN	<u>+MaBN(PK)[PK]</u> <u>+SoCMND[PK]</u> Hoten SDT Diachi Gioitinh Chuyendentu Thongtinxnbandau <i>Ma_Staff(FK)</i> Ngaynhapvien Ngayxuatvien <i>Ma_Nurse(FK)</i>
2. BENH LYDIKEM	<u>+MaBN [PK]</u> <u>+Benhlydikem [PK]</u>
3. TRIEU CHUNG THUONG	<u>+MaBN[PK]</u> <u>+ten_trieu_chung_thuong[PK]</u>
4. TRIEU CHUNG NGHIEM TRONG	<u>+MaBN[PK]</u> <u>+ten_trieu_chung_nghiem_trong[PK]</u>
5. XET NGHIEM	<u>+MaXN[PK]</u> <u>+MaBN[PK]</u> ThoigianXN PCR_result SPO2 Respiratory Rate QT_result ct

	Warning
6. PHONG	<u>+MaPhong[PK]</u> tang loaiphong toanha succhua
7. Staff_PhanVao_Phong	<u>+MaPhong [PK]</u> <u>+Ma_Staff [PK]</u> <u>+MaBN [PK]</u> Tinhtranghientai VittriBN dateChange

Bảng 1: Bảng thuộc tính của thực thể và mối quan hệ

3) Bảng thể hiện Datatype của CSDL

3.1) Datatype của thực thể BỆNH NHÂN

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaBN	Int(6)	PK	Mã Bệnh Nhân
HoTen	Varchar(50)		Họ tên
SoCMND	Char(12)	PK	Số CMND
SDT	Char(10)		Số điện thoại
Diachi	Varchar(255)		Địa chỉ
Gioitinh	Char(1)		Giới tính (M-F)
Chuyendentu	Varchar(255)		Chuyên đến từ đâu?
Thongtinxnbandau	Varchar(255)		Thông tin xét nghiệm ban đầu
Ma_Staff	Int(6)	FK	Mã Staff tiếp nhận
Ngaynhapvien	Datetime		Ngày nhập viện
Ngayxuatvien	datetime		Ngày xuất viện
Ma_Nurse	Int(6)	FK	Mã Nurse chăm sóc

Bảng 2: Bảng kiểu dữ liệu của table BỆNH NHÂN

3.2) Datatype của thực thể BỆNH LÝ ĐI KÈM

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaBN	Int(6)	PK	Mã bệnh nhân
benhlydikem	Varchar(100)	PK	Bệnh lý đi kèm

Bảng 3: Bảng kiểu dữ liệu của table BỆNH LÝ ĐI KÈM

3.3) Datatype của thực thể TRIỆU CHỨNG THƯỜNG

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaBN	Int(6)	PK	Mã bệnh nhân
Ten_trieu_chung_thuong	Varchar(100)	PK	Triệu chứng thường

Bảng 4: Bảng kiểu dữ liệu của table TRIỆU CHỨNG THƯỜNG

3.4) Datatype của thực thể TRIỆU CHỨNG NGHIÊM TRỌNG

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaBN	Int(6)	PK	Mã bệnh nhân
Ten_trieu_chung_nghiem_trong	Varchar(100)	PK	Triệu chứng nghiêm trọng

Bảng 5: Bảng kiểu dữ liệu của table TRIỆU CHỨNG NGHIÊM TRỌNG

3.5) Datatype của thực thể XÉT NGHIỆM

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaXN	Int(3)	PK	Mã xét nghiệm
MaBN	Int(6)	PK	Mã bệnh nhân
ThoigianXN	datetime		Thời gian xét nghiệm
PCR_result	Char(1)		Kết quả của xét nghiệm PCR (T-F)
SPO2	Decimal(3,2)		Giá trị mức bão hòa oxy trong máu
Respiratory_Rate	Int(3)		Nhịp thở của bệnh nhân
QT_result	Char(1)		Kết quả của xét nghiệm Quick Test(T-F)
ct	Int(3)		Giá trị ngưỡng chu kì (ct) của xét nghiệm Quick Test
Warning	Char(1)		Trạng thái cảnh báo dựa vào kết quả xét nghiệm. Đánh dấu bằng 1 kí tự “x”.

Bảng 6: Bảng kiểu dữ liệu của table XÉT NGHIỆM

3.6) Datatype của thực thể PHÒNG

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaPhong	Int(6)	PK	Mã phòng
tang	Int(2)		Tầng
loaiphong	Varchar(10)		Loại phòng
toanha	Varchar(3)		Tòa nhà
succhua	Int(2)		Sức chứa

Bảng 7: Bảng kiểu dữ liệu của table PHÒNG

3.7) Datatype của thực thể Staff_PhanVao_Phong

Tên trường	Kiểu dữ liệu	Khóa	Ghi chú
MaPhong	Int(6)	PK	Mã phòng
Ma_Staff	Int(6)	PK	Mã staff phân vào phòng
MaBN	Int(6)	PK	Mã bệnh nhân được staff phân vào phòng
Tinh_trang_hien_tai	Varchar(50)		Tình trạng bệnh nhân
Vitri_BN	Varchar(50)		Vị trí hiện tại của bệnh nhân
dateChange	Datetime		Ngày/Giờ chuyển BN

Bảng 8: Bảng kiểu dữ liệu của table Staff_PhanVao_Phong

4) Demo data cho các tables.

4.1) Demo data cho table BENH NHAN

MaBN	HoTen	SoCMND	SDT	DiaChi	Gioiti nh	chuyend entu	thongtinxnbau	Ma_Sta ff	Ngày_nhap_vien	Ngày_xuat_vie n	Ma_Nu rse
1	Nguyễn Văn A	123456789012	987654321	Hà Nội	M	Bệnh viện huyện	Dương tính	1	2023-10-01	null	2
2	Trần Thị B	123456789013	987654322	Đà Nẵng	F	Sân bay	Dương tính	2	2023-10-05	2023-10-10	3
3	Lê Văn C	123456789014	987654323	TP. Hồ Chí Minh	M	Sân bay	Dương tính	3	2023-10-10	null	1
4	Phạm Thị D	123456789015	987654324	Cần Thơ	F	Khu cách ly quận 7	Dương tính	4	2023-10-15	2023-10-20	2
5	Huỳnh Văn E	123456789	98765	Hải	M	Quận 6	Dương tính	1	2023-10-20	2023-10-25	3

Bảng 9: Bảng kiểu demo data của table BỆNH NHÂN

		016	4325	Phòng							
--	--	-----	------	-------	--	--	--	--	--	--	--

4.2) Demo data cho table BenhLyDiKem

MaBN	benhlydikem
1	Đau bụng
1	Viêm loét dạ dày
2	Sốt cao
2	Cảm lạnh
3	Chấn thương sọ não
3	Gãy xương
4	Ung thư
4	Ra máu sau sinh
5	Viêm phổi
5	Suy hô hấp

Bảng 10: Bảng kiểu demo data của table BenhLyDiKem

4.3) Demo data cho table TrieuChungThuong

MaBN	ten_trieu_chung_thuong
1	Đau bụng dữ dội
1	Nôn mửa
2	Sốt cao trên 39 độ C
2	Ho nhiều
3	Đau đầu dữ dội
3	Chảy máu mũi
4	Chảy máu mũi
4	Đau bụng dưới
5	Ho ra đờm
5	Khó thở

Bảng 11: Bảng kiểu demo data của table TrieuChungThuong

4.4) Demo data cho table TrieuChungNghiemTrong

MaBN	ten_trieu_chung_nghiem_trong
1	Đau bụng dữ dội kèm theo nôn mửa
1	Sốt cao trên 39 độ C kèm theo ho nhiều
2	Đau đầu dữ dội kèm theo chảy máu mũi
2	Sốt cao trên 39 độ C kèm theo ho nhiều
3	Ho ra đờm có màu xanh hoặc vàng
3	Khó thở kèm theo tím tái môi, đầu ngón tay

Bảng 12: Bảng kiểu demo data của table TrieuChungNghiemTrong

4.5) Demo data cho table XÉT NGHIỆM

MaXN	MaBN	ThoigianXN	PCR_result	SPO2	Respiratory_Rate	QT_result	ct	Warning
1	1	2023-10-03	T	97.5	20	F	0	0
2	2	2023-10-04	F	95.2	16	F	0	0
3	3	2023-10-05	F	98.1	18	F	0	0
4	4	2023-10-06	T	96.8	22	T	23	1
5	5	2023-10-07	T	99.3	20	T	33	0
6	1	2023-10-10	T	96.7	24	T	35	0
7	2	2023-10-11	F	98.4	18	F	0	0
8	3	2023-10-12	F	99.2	20	T	30	0
9	4	2023-10-13	F	97.1	22	F	0	1
10	5	2023-10-14	T	98.9	24	F	0	0

Bảng 13: Bảng kiểu demo data của table XetNghiem

4.6) Demo data cho table PHÒNG

MaPhong	tang	loaiphong	toanha	succhua
1	1	Phòng thường	A	25
2	2	Phòng cấp cứu	B	25
3	3	Phòng hồi sức	C	25

4	4	Phòng thường	A	25
5	5	Phòng cấp cứu	B	25
6	6	Phòng hồi sức	C	25

Bảng 14: Bảng kiểu demo data của table PHONG

4.7) Demo data cho table Staff_PhanVao_Phong

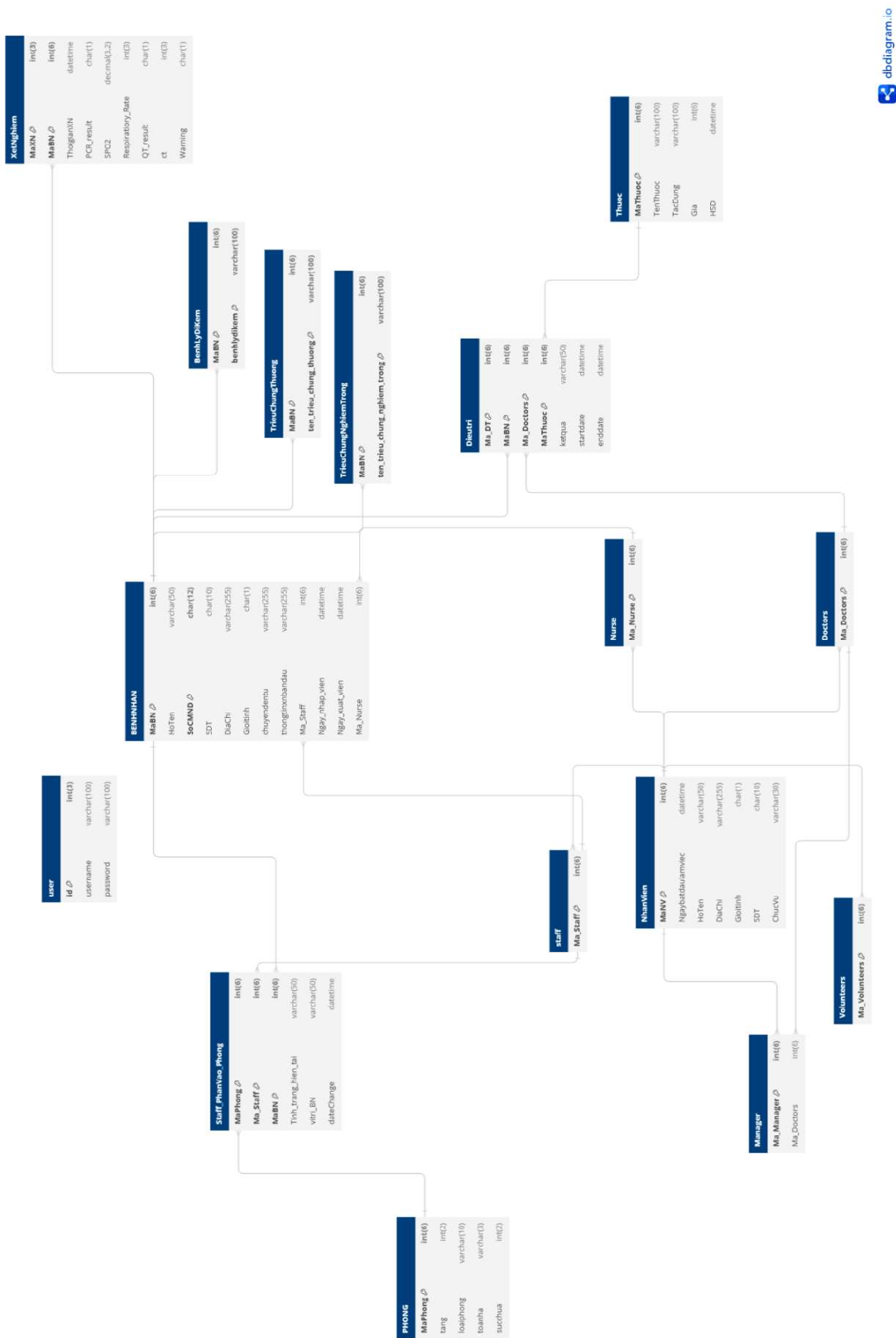
MaPhong	Ma_Staff	MaBN	Tinh_trang_hien_tai	vitri_BN	dateChange
1	1	1	Đang theo dõi	Phòng thường	2023-12-02 12:21:00
2	2	2	Khó thở	Phòng cấp cứu	2023-12-02 12:22:00
3	3	3	Hồi phục	Phòng hồi sức	2023-12-02 12:23:00
4	4	4	Đang theo dõi	Phòng thường	2023-12-02 12:24:00
5	5	5	Sốt cao	Phòng cấp cứu	2023-12-02 12:25:00
6	6	6	Hồi phục	Phòng hồi sức	2023-12-02 15:25:00

Bảng 15: Bảng kiểu demo data của table Staff_PhanVao_Phong

5) Relational Mapping

Realational Data Mapping (Database Diagram) tổng quan cho toàn hệ thống: Để ra được RDM này thì em dùng công cụ dbdiagram.io để vẽ và mapping các tables lại với nhau.

(hình trang sau)



Hình 13: Relational Data Mapping

V/ SQL COMMANDS

1) Create Table

1.1) Create Table BENHNHAN

```
1. CREATE TABLE `BENHNHAN` (  
2.   `MaBN` int(6),  
3.   `HoTen` varchar(50),  
4.   `SoCMND` char(12),  
5.   `SDT` char(10),  
6.   `DiaChi` varchar(255),  
7.   `Gioitinh` char(1),  
8.   `chuyendentu` varchar(255),  
9.   `thongtinxnbandau` varchar(255),  
10.  `Ma_Staff` int(6),  
11.  `Ngay_nhap_vien` datetime,  
12.  `Ngay_xuat_vien` datetime,  
13.  `Ma_Nurse` int(6),  
14.  PRIMARY KEY (`MaBN`, `SoCMND` )  
15. );
```

- **MaBN**: Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **HoTen**: Đây là cột varchar lưu trữ họ tên đầy đủ của bệnh nhân. Độ dài tối đa của cột này là 50 ký tự.

- **SoCMND**: Đây là cột char lưu trữ số CMND của bệnh nhân. Độ dài tối đa của cột này là 12 ký tự.

- **SDT**: Đây là cột char lưu trữ số điện thoại của bệnh nhân. Độ dài tối đa của cột này là 10 ký tự.

- **DiaChi**: Đây là cột varchar lưu trữ địa chỉ của bệnh nhân. Độ dài tối đa của cột này là 255 ký tự.

- **Gioitinh**: Đây là cột char lưu trữ giới tính của bệnh nhân. Độ dài tối đa của cột này là 1 ký tự.

- **chuyendentu**: Đây là cột varchar lưu trữ tên bệnh viện hoặc phòng khám mà bệnh nhân chuyển đến. Độ dài tối đa của cột này là 255 ký tự.

- **thongtinxbandau**: Đây là cột varchar lưu trữ thông tin y tế ban đầu của bệnh nhân. Độ dài tối đa của cột này là 255 ký tự.

- **Ma_Staff**: Đây là cột số nguyên lưu trữ mã staff tiếp nhận bệnh nhân. Độ dài tối đa của cột này là 6 chữ số.

- **Ngay_nhap_vien**: Đây là cột datetime lưu trữ ngày và giờ bệnh nhân nhập viện.

- **Ngay_xuat_vien**: Đây là cột datetime lưu trữ ngày và giờ bệnh nhân xuất viện.

- **Ma_Nurse**: Đây là cột số nguyên lưu trữ mã nurse chăm sóc bệnh nhân. Độ dài tối đa của cột này là 6 chữ số.

1.2) Create Table BenhLyDiKem

```
1. CREATE TABLE `BenhLyDiKem` (  
2.   `MaBN` int(6),  
3.   `benhlydikem` varchar(100),  
4.   PRIMARY KEY (`MaBN`, `benhlydikem`)  
5. );
```

- **MaBN**: Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **benhlydikem**: Cột varchar lưu bệnh lý đi kèm của bệnh nhân. Độ dài tối đa 100 ký tự.

1.3) Create Table TrieuChungThuong

```
1. CREATE TABLE `TrieuChungThuong` (  
2.   `MaBN` int(6),  
3.   `ten_trieu_chung_thuong` varchar(100),  
4.   PRIMARY KEY (`MaBN`, `ten_trieu_chung_thuong`)  
5. );
```

- **MaBN**: Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **ten_trieu_chung_thuong**: Cột varchar lưu triệu chứng thường của bệnh nhân. Độ dài tối đa 100 ký tự.

1.4) Create Table TrieuChungNghiemTrong

```
1. CREATE TABLE `TrieuChungNghiemTrong` (  
2.   `MaBN` int(6),  
3.   `ten_trieu_chung_nghiem_trong` varchar(100),  
4.   PRIMARY KEY (`MaBN`, `ten_trieu_chung_nghiem_trong`)  
5. );
```

- **MaBN**: Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **ten_trieu_chung_nghiem_trong**: Cột varchar lưu triệu chứng nghiêm trọng của bệnh nhân. Độ dài tối đa 100 ký tự.

1.5) Create Table XETNGHIEM

```
1. CREATE TABLE `XetNghiem` (  
2.   `MaXN` int(3),  
3.   `MaBN` int(6),  
4.   `ThoigianXN` datetime,  
5.   `PCR_result` char(1),  
6.   `SP02` decimal(3,2),  
7.   `Respiratory_Rate` int(3),  
8.   `QT_result` char(1),  
9.   `ct` int(3),  
10.  `Warning` char(1),  
11.  PRIMARY KEY (`MaXN`, `MaBN`)  
12. );
```

- **MaXN**: Số nguyên có độ dài tối đa 3 ký tự. Cột này lưu trữ ID 1 đợt xét nghiệm của bệnh nhân.

- **MaBN:** Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **ThoigianXN:** Cột datetime để lưu trữ thời gian xét nghiệm.

- **PCR_result:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ kết quả của xét nghiệm PCR. (kết quả là dương tính(T) hoặc âm tính (F))

- **SPO2:** Cột số thập phân có format như sau: 123.45. Cột này có thể lưu trữ mức bão hòa oxy trong máu (SPO2) của bệnh nhân.

- **Respiratory_Rate:** Số nguyên có độ dài tối đa 3 ký tự. Lưu trữ nhịp thở của bệnh nhân.

- **QT_result:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ kết quả của xét nghiệm Quick Test. (kết quả là dương tính(T) hoặc âm tính (F))

- **ct:** Cột số nguyên để lưu trữ giá trị ngưỡng chu kỳ (ct).

- **Warning:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ một dấu hiệu cho biết liệu có bất kỳ cảnh báo nào liên quan đến kết quả xét nghiệm của bệnh nhân hay không.

1.6) Create Table PHONG

```
1. CREATE TABLE `PHONG` (  
2.   `MaPhong` int(6) PRIMARY KEY,  
3.   `tang` int(2),  
4.   `loaiphong` varchar(10),  
5.   `toanha` varchar(3),  
6.   `succhua` int(2)  
7. );
```

- **MaPhong:** Cột số nguyên lưu mã phòng. Độ dài tối đa 6 ký tự.

- **tang:** Cột số nguyên lưu tầng của phòng. Độ dài tối đa 2 ký tự.

- **loaiphong:** Cột varchar lưu loại phòng. Độ dài tối đa 10 ký tự.

- **toanha:** Cột varchar lưu toàn nhà của phòng. Độ dài tối đa 3 ký tự.

- **succhua:** Cột số nguyên lưu sức chứa của phòng. Độ dài tối đa 2 ký tự.

1.7) Create Table Staff_PhanVao_Phong

```
1. CREATE TABLE `Staff_PhanVao_Phong` (  
2.   `MaPhong` int(6),  
3.   `Ma_Staff` int(6),  
4.   `MaBN` int(6),  
5.   `Tinh_trang_hien_tai` varchar(50),  
6.   `vitri_BN` varchar(50),  
7.   `dateChange` datetime,  
8.   PRIMARY KEY (`MaPhong`, `Ma_Staff`, `MaBN`)  
9. );
```

- **MaPhong:** Cột số nguyên lưu mã phòng. Độ dài tối đa 6 kí tự.
- **Ma_Staff:** Cột số nguyên lưu mã staff phân vào phòng. Độ dài tối đa 6 kí tự.
- **MaBN:** Cột số nguyên lưu mã bệnh nhân được staff phân vào phòng. Độ dài tối đa 6 kí tự.
- **ting_trang_hien_tai:** Cột varchar lưu tình trạng của bệnh nhân. Độ dài tối đa 50 kí tự.
- **vitri_BN:** Cột số varchar lưu vị trí hiện tại của bệnh nhân. Độ dài tối đa 50 kí tự.
- **dateChange:** Cột datetime lưu ngày giờ chuyển phòng

Ngoài việc create table được phân công thì em còn viết câu create cho các table sau:

1.8) Create Table XetNghiem

```
1. CREATE TABLE `XetNghiem` (  
2.   `MaXN` int(3),  
3.   `MaBN` int(6),  
4.   `ThoigianXN` datetime,  
5.   `PCR_result` char(1),  
6.   `SP02` decimal(3,2),  
7.   `Respiratory_Rate` int(3),  
8.   `QT_result` char(1),  
9.   `ct` int(3),
```

```

10.     `Warning` char(1),
11.     PRIMARY KEY (`MaBN`, `MaXN`)
12. );

```

- **MaXN:** Số nguyên có độ dài tối đa 3 ký tự. Cột này lưu trữ ID 1 đợt xét nghiệm của bệnh nhân.

- **MaBN:** Số nguyên có độ dài tối đa 6 ký tự. Cột này có thể lưu trữ mã bệnh nhân.

- **ThoigianXN:** Cột datetime để lưu trữ thời gian xét nghiệm.

- **PCR_result:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ kết quả của xét nghiệm PCR. (kết quả là dương tính(T) hoặc âm tính (F))

- **SPO2:** Cột số thập phân có format như sau: 123.45. Cột này có thể lưu trữ mức bão hòa oxy trong máu (SPO2) của bệnh nhân.

- **Respiratory_Rate:** Số nguyên có độ dài tối đa 3 ký tự. Lưu trữ nhịp thở của bệnh nhân.

- **QT_result:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ kết quả của xét nghiệm Quick Test. (kết quả là dương tính(T) hoặc âm tính (F))

- **ct:** Cột số nguyên để lưu trữ giá trị ngưỡng chu kỳ (ct) của Quick Test.

- **Warning:** Cột ký tự có độ dài tối đa 1 ký tự. Cột này có thể lưu trữ một dấu hiệu cho biết liệu có bất kỳ cảnh báo nào liên quan đến kết quả xét nghiệm của bệnh nhân hay không.

1.9) Create Table Phong

```

1. CREATE TABLE `PHONG` (
2.     `MaPhong` int(6) PRIMARY KEY,
3.     `tang` int(2),
4.     `loaiphong` varchar(10),
5.     `toanha` varchar(3),
6.     `succhua` int(2)
6. );

```

- **MaPhong:** Cột số nguyên lưu mã phòng. Độ dài tối đa 6 ký tự.

- **tang:** Cột số nguyên lưu tầng của phòng. Độ dài tối đa 2 ký tự.

- **loaiphong:** Cột varchar lưu loại phòng. Độ dài tối đa 10 ký tự.

- **toanha:** Cột varchar lưu toàn nhà của phòng. Độ dài tối đa 3 kí tự.
- **succhua:** Cột số nguyên lưu sức chứa của phòng. Độ dài tối đa 2 kí tự.

1.10) Create Table NhanVien

```
1. CREATE TABLE `NhanVien` (
2.   `MaNV` int(6) PRIMARY KEY,
3.   `Ngaybatdaulamviec` datetime,
4.   `HoTen` varchar(50),
5.   `DiaChi` varchar(255),
6.   `Gioitinh` char(1),
7.   `SDT` char(10),
8.   `ChucVu` varchar(30)
9. );
```

- **MaNV:** Cột số nguyên lưu mã nhân viên. Độ dài tối đa 6 kí tự.
- **Ngaybatdaulamviec:** Cột datetime để lưu ngày bắt đầu làm việc của nhân viên.
- **HoTen:** Cột varchar lưu họ tên nhân viên. Độ dài tối đa 50 kí tự.
- **DiaChi:** Cột varchar lưu địa chỉ nhân viên. Độ dài tối đa 255 kí tự.
- **Gioitinh:** Cột char lưu giới tính nhân viên (M/F). Độ dài tối đa 1 kí tự.
- **SDT:** Cột char lưu số điện thoại nhân viên. Độ dài tối đa 10 kí tự.
- **ChucVu:** Cột varchar lưu chức vụ nhân viên. Độ dài tối đa 30 kí tự.

1.11) Create Table staff, Doctors, Nurse, Manager, Volunteers

```
1. CREATE TABLE `staff` (
2.   `Ma_Staff` int(6) PRIMARY KEY
3. );
4. CREATE TABLE `Doctors` (
5.   `Ma_Doctors` int(6) PRIMARY KEY
6. );
7. CREATE TABLE `Nurse` (
8.   `Ma_Nurse` int(6) PRIMARY KEY
9. );
10. CREATE TABLE `Manager` (
11.   `Ma_Manager` int(6),
```



```

12.  `Ma_Doctors` int(6),
13.  PRIMARY KEY (`Ma_Manager`, `Ma_Doctors`)
14. );
15. CREATE TABLE `Volunteers` (
16.  `Ma_Volunteers` int(6) PRIMARY KEY
17. );

```

- **Ma_Staff:** Cột số nguyên lưu mã Staff. Độ dài tối đa 6 kí tự.
- **Ma_Doctors:** Cột số nguyên lưu mã Doctors. Độ dài tối đa 6 kí tự.
- **Ma_Nurse:** Cột số nguyên lưu mã Nurse. Độ dài tối đa 6 kí tự.
- **Ma_Manager:** Cột số nguyên lưu mã Manager. Độ dài tối đa 6 kí tự.
- **Ma_Doctors:** Cột số nguyên lưu mã Doctors được chỉ định làm manager. Độ dài tối đa 6 kí tự.
- **Ma_Volunteers:** Cột số nguyên lưu mã Volunteers. Độ dài tối đa 6 kí tự.

1.12) Create Table Thuoc

```

1. CREATE TABLE `Thuoc` (
2.  `MaThuoc` int(6) PRIMARY KEY,
3.  `TenThuoc` varchar(100),
4.  `TacDung` varchar(100),
5.  `Gia` int(6),
6.  `HSD` datetime
7. );

```

- **MaThuoc:** Cột số nguyên lưu mã thuốc. Độ dài tối đa 6 kí tự.
- **TenThuoc:** Cột varchar để lưu tên thuốc. Độ dài tối đa 100 kí tự.
- **TacDung:** Cột varchar lưu tác dụng của thuốc. Độ dài tối đa 100 kí tự.
- **Gia:** Cột số nguyên lưu giá tiền của thuốc. Độ dài tối đa 6 kí tự.
- **HSD:** Cột datetime lưu thời hạn sử dụng của thuốc.

1.13) Create Table Dieutri

```

1. CREATE TABLE `Dieutri` (
2.  `Ma_DT` int(6),
3.  `Ma_Doctors` int(6),
4.  `MaThuoc` int(6),

```

```

5.     `MaBN` int(6),
6.     `ngay_batdau_dieutri` datetime,
7.     `ngay_ketthuc_dieutri` datetime,
8.     `ketqua` varchar(100),
9.     PRIMARY KEY (`Ma_Doctors`, `MaThuoc`, `MaBN`)
10. );

```

- **Ma_DT:** Cột số nguyên lưu mã của cuộc điều trị. Độ dài tối đa 6 kí tự.

- **Ma_Doctors:** Cột số nguyên lưu mã Doctors tham gia vào cuộc điều trị. Độ dài tối đa 6 kí tự.

- **MaThuoc:** Cột số nguyên để lưu mã thuốc tham gia vào cuộc điều trị. Độ dài tối đa 6 kí tự

- **MaBN:** Cột số nguyên để lưu mã bệnh nhân tham gia vào cuộc điều trị. Độ dài tối đa 6 kí tự

- **ngay_batdau_dieutri:** Cột datetime lưu ngày bắt đầu điều trị.

- **ngay_ketthuc_dieutri:** Cột datetime lưu ngày kết thúc điều trị.

- **ketqua:** Cột varchar lưu kết quả của cuộc điều trị. Độ dài tối đa 100 kí tự.

1.14) Create Table user

```

1. CREATE TABLE `user` (
2.     `id` int(3) PRIMARY KEY,
3.     `username` varchar(100),
4.     `password` varchar(100));

```

- **id:** Cột số nguyên lưu mã user. Độ dài tối đa 1 kí tự.

- **username:** Cột varchar lưu username. Độ dài tối đa 100 kí tự.

- **password:** Cột varchar lưu password. Độ dài tối đa 100 kí tự.

2) Insert

2.1) Insert into Table BENHNHAN

```

1. INSERT INTO BENHNHAN
(MaBN, HoTen, SoCMND, SDT, DiaChi, Gioitinh, chuyendentu, thongtinxnbandau, Ma_Staff, Ngay_nhap_vien, Ngay_xuat_vien, Ma_Nurse)
2. VALUES (1, 'Nguyễn Văn A', '123456789012', '0987654321', 'Hà Nội', 'M', 'Bệnh viện huyện', 'Dương tính', 1, '2023-10-01', 'null', 2),

```

```

3. (2, 'Trần Thị B', '123456789013', '0987654322', 'Đà Nẵng', 'F', 'Phòng khám tư', 'Dương tính', 2, '2023-10-05', '2023-10-10', 3),
4. (3, 'Lê Văn C', '123456789014', '0987654323', 'TP. Hồ Chí Minh', 'M', 'Trạm y tế xã', 'Dương tính', 3, '2023-10-10', 'null', 1),
5. (4, 'Phạm Thị D', '123456789015', '0987654324', 'Cần Thơ', 'F', 'Bệnh viện đa khoa tỉnh', 'Dương tính', 4, '2023-10-15', '2023-10-20', 2),
6. (5, 'Huỳnh Văn E', '123456789016', '0987654325', 'Hải Phòng', 'M', 'Phòng khám tư', 'Dương tính', 1, '2023-10-20', '2023-10-25', 3);

```

2.2) Insert into Table BenhLyDiKem

```

1. INSERT INTO BenhLyDiKem (MaBN, benhlydikem)
2. VALUES
3. (1, 'Đau bụng'),
4. (1, 'Viêm loét dạ dày'),
5. (2, 'Sốt cao'),
6. (2, 'Cảm lạnh'),
7. (3, 'Chấn thương sọ não'),
8. (3, 'Gãy xương'),
9. (4, 'Ung thư'),
10. (4, 'Ra máu sau sinh'),
11. (5, 'Viêm phổi'),
12. (5, 'Suy hô hấp');

```

2.3) Insert into Table TrieuChungThuong

```

1. INSERT INTO TrieuChungThuong (MaBN, ten_trieu_chung_thuong)
2. VALUES (1, 'Đau bụng dữ dội'),
3. (1, 'Nôn mửa'),
4. (2, 'Sốt cao trên 39 độ C'),
5. (2, 'Ho nhiều'),

```

```

6. (3, 'Đau đầu dữ dội'),
7. (3, 'Chảy máu mũi'),
8. (4, 'Chảy máu mũi'),
9. (4, 'Đau bụng dưới'),
10. (5, 'Ho ra đờm'),
11. (5, 'Khó thở');

```

2.4) Insert into Table TrieuChungNghiemTrong

```

1. INSERT INTO TrieuChungNghiemTrong
(MaBN, ten_trieu_chung_nghiem_trong)
2. VALUES (1, 'Đau bụng dữ dội kèm theo nôn mửa'),
3. (1, 'Sốt cao trên 39 độ C kèm theo ho nhiều'),
4. (2, 'Đau đầu dữ dội kèm theo chảy máu mũi'),
5. (2, 'Sốt cao trên 39 độ C kèm theo ho nhiều'),
6. (3, 'Ho ra đờm có màu xanh hoặc vàng'),
7. (3, 'Khó thở kèm theo tím tái môi, đầu ngón tay');

```

2.5) Insert into Table XetNghiem

```

1. INSERT INTO XetNghiem
(MaXN, MaBN, ThoigianXN, PCR_result, SPO2, Respiratory_Rate, QT_result, ct, Warning)
2. VALUES
3. (1, 1, '2023-10-03', 'T', 97.5, 20, 'F', 0, 0),
4. (2, 2, '2023-10-04', 'F', 95.2, 16, 'F', 0, 0),
5. (3, 3, '2023-10-05', 'F', 98.1, 18, 'F', 0, 0),
6. (4, 4, '2023-10-06', 'T', 96.8, 22, 'T', 23, 1),
7. (5, 5, '2023-10-07', 'T', 99.3, 20, 'T', 33, 0),
8. (6, 1, '2023-10-10', 'T', 96.7, 24, 'T', 35, 0),
9. (7, 2, '2023-10-11', 'F', 98.4, 18, 'F', 0, 0),
10. (8, 3, '2023-10-12', 'F', 99.2, 20, 'T', 30, 0),
11. (9, 4, '2023-10-13', 'F', 97.1, 22, 'F', 0, 1),

```

```
12. (10, 5, '2023-10-14', 'T', 98.9, 24, 'F', 0, 0);
```

2.6) Insert into Table Phong

```
1. INSERT INTO PHONG (MaPhong,tang,loai phong,toan ha,suc chua)
2. VALUES
3. (1, 1, 'Phòng thường', 'A', 25),
4. (2, 2, 'Phòng cấp cứu', 'B', 25),
5. (3, 3, 'Phòng hồi sức', 'C', 25),
6. (4, 4, 'Phòng thường', 'A', 25),
7. (5, 5, 'Phòng cấp cứu', 'B', 25),
8. (6, 6, 'Phòng hồi sức', 'C', 25);
```

2.7) Insert into Table Staff_PhanVao_Phong

```
1. INSERT INTO Staff_PhanVao_Phong
(MaPhong,Ma_Staff,MaBN,Tinh_trang_hien_tai,vitri_BN)
2. VALUES
3. (1, 1, 1, 'Đang theo dõi', 'Phòng thường','2023-12-02 12:21:00'),
4. (2, 2, 2, 'Khó thở', 'Phòng cấp cứu', '2023-12-02 12:22:00'),
5. (3, 3, 3, 'Hồi phục', 'Phòng hồi sức', '2023-12-02 12:23:00'),
6. (4, 4, 4, 'Đang theo dõi', 'Phòng thường', '2023-12-02 12:24:00'),
7. (5, 5, 5, 'Sốt cao', 'Phòng cấp cứu', '2023-12-02 12:25:00'),
8. (6, 6, 6, 'Hồi phục', 'Phòng hồi sức', '2023-12-02 15:25:00');
```

Ngoài những table được phân công thì em có làm luôn những table khác bao gồm Table: nhanvien, staff, nurse, manager, doctors, volunteer, phong, thuoc.

2.8) Insert into nhanvien

```
1. INSERT INTO `nhanvien` (`MaNV`, `Ngaybatdaulamviec`, `HoTen`, `DiaChi`,
`Gioitinh`, `SDT`, `ChucVu`) VALUES
2. (1, '2023-12-11 10:00:00', 'Nguyễn Văn Kiên', '123 Nguyễn Văn Linh, Ho Chi Minh City', 'M', '0987654321', 'Staff'),
3. (2, '2023-12-12 12:00:00', 'Trần Thị Linh', '456 Lê Văn Lương, Ho Chi Minh City', 'F', '0123456789', 'Staff');
```

4. (3, '2023-12-13 14:00:00', 'Lê Văn Minh', '789 Nguyễn Đình Chiểu, Ho Chi Minh City', 'M', '0901234567', 'Staff'),
5. (4, '2023-12-14 16:00:00', 'Phạm Thị Như', '1011 Nguyễn Huệ, Ho Chi Minh City', 'F', '0876543210', 'Staff'),
6. (5, '2023-12-15 18:00:00', 'Huỳnh Văn Tuấn', '1213 Nguyễn Thông, Ho Chi Minh City', 'M', '0789012345', 'Staff'),
7. (6, '2023-12-06 10:00:00', 'Nguyễn Thị Phước', '123 Nguyễn Văn Linh, Ho Chi Minh City', 'M', '0987654321', 'Nurse'),
8. (7, '2023-12-07 12:00:00', 'Trần Văn Hải', '456 Lê Văn Lương, Ho Chi Minh City', 'F', '0123456789', 'Nurse'),
9. (8, '2023-12-08 14:00:00', 'Lê Thị Hòa', '789 Nguyễn Đình Chiểu, Ho Chi Minh City', 'M', '0901234567', 'Nurse'),
10. (9, '2023-12-09 16:00:00', 'Phạm Văn Yên', '1011 Nguyễn Huệ, Ho Chi Minh City', 'F', '0876543210', 'Nurse'),
11. (10, '2023-12-10 18:00:00', 'Huỳnh Thị Linh', '1213 Nguyễn Thông, Ho Chi Minh City', 'M', '0789012345', 'Nurse'),
12. (11, '2023-12-01 10:00:00', 'Nguyễn Văn Anh', '123 Nguyễn Văn Linh, Ho Chi Minh City', 'M', '0987654321', 'Manager'),
13. (12, '2023-12-02 12:00:00', 'Trần Thị Bình', '456 Lê Văn Lương, Ho Chi Minh City', 'F', '0123456789', 'Manager'),
14. (13, '2023-12-03 14:00:00', 'Lê Văn Chương', '789 Nguyễn Đình Chiểu, Ho Chi Minh City', 'M', '0901234567', 'Manager'),
15. (14, '2023-12-04 16:00:00', 'Phạm Thị Diện', '1011 Nguyễn Huệ, Ho Chi Minh City', 'F', '0876543210', 'Manager'),
16. (15, '2023-12-01 21:27:40', 'Do Phúc Thuận', 'Viet Nam', 'M', '0918839272', 'Manager'),
17. (16, '2023-12-10 18:00:00', 'Phạm Đình Thịnh', '123 Nguyễn Văn Linh, Ho Chi Minh City', 'M', '0987654321', 'Doctor'),
18. (17, '2023-12-06 18:00:00', 'Ngô Diệc Phàm', '456 Lê Văn Lương, Ho Chi Minh City', 'F', '0123456789', 'Doctor'),
19. (18, '2023-12-07 18:00:00', 'Hoàng Văn Ớn', '789 Nguyễn Đình Chiểu, Ho Chi Minh City', 'M', '0901234567', 'Doctor'),
20. (19, '2023-12-08 18:00:00', 'Nguyễn Tiến Đạt', '1011 Nguyễn Huệ, Ho Chi Minh City', 'F', '0876543210', 'Doctor'),
21. (20, '2023-12-09 18:00:00', 'Trần Thị Yến Lê', '1213 Nguyễn Thông, Ho Chi Minh City', 'M', '0789012345', 'Doctor'),
22. (21, '2023-01-27 18:00:00', 'Lương Bằng', '123 Nguyễn Văn Linh, Ho Chi Minh City', 'M', '0987654321', 'Volunteer'),
23. (22, '2023-12-11 18:00:00', 'Trần Hoàng Thanh Yến', '456 Lê Văn Lương, Ho Chi Minh City', 'F', '0123456789', 'Volunteer'),
24. (23, '2023-12-12 18:00:00', 'Võ Khắc Toàn', '789 Nguyễn Đình Chiểu, Ho Chi Minh City', 'M', '0901234567', 'Volunteer'),
25. (24, '2023-12-13 18:00:00', 'Nguyễn Trần Hoàng Xuân', '1011 Nguyễn Huệ, Ho Chi Minh City', 'F', '0876543210', 'Volunteer'),
26. (25, '2023-12-14 18:00:00', 'Hồ Văn Cường', '1213 Nguyễn Thông, Ho Chi Minh City', 'M', '0789012345', 'Volunteer'),
27. (26, '2023-12-15 18:00:00', 'Lê Trung Minh', '1213 Nguyễn Thông, Ho Chi Minh City', 'M', '0789012345', 'Volunteer');

2.9) Insert into staff

```
1. INSERT INTO `staff` (`Ma_Staff`) VALUES
2. (1),
3. (2),
4. (3),
5. (4),
6. (5);
```

2.10) Insert into nurse

```
1. INSERT INTO `nurse` (`Ma_Nurse`) VALUES
2. (6),
3. (7),
4. (8),
5. (9),
6. (10);
```

2.12) Insert into manager

```
1. INSERT INTO `manager` (`Ma_Manager`, `Ma_Doctors`) VALUES
2. (11, NULL),
3. (12, NULL),
4. (13, NULL),
5. (14, NULL),
6. (15, NULL),
7. (17, 17);
```

2.13) Insert into doctors

```
1. INSERT INTO `doctors` (`Ma_Doctors`) VALUES
2. (16),
3. (17),
4. (18),
5. (19),
6. (20);
```

2.14) Insert into volunteers

```
1. INSERT INTO `volunteers` (`Ma_Volunteers`) VALUES
2. (21),
3. (22),
4. (23),
5. (24),
6. (25),
7. (26);
```

2.15) Insert into table phong

```
1. INSERT INTO `phong` (`MaPhong`, `tang`, `loaiphong`, `toanha`, `succhua`)
VALUES
2. (1, 1, 'PT', 'A', 12),
3. (2, 1, 'PT', 'A', 12),
4. (3, 1, 'CC', 'A', 7),
5. (4, 1, 'CC', 'A', 7),
6. (5, 2, 'PT', 'A', 12),
7. (6, 2, 'HS', 'A', 5),
8. (7, 2, 'HS', 'A', 5),
9. (8, 3, 'CC', 'A', 5),
10. (9, 3, 'PT', 'A', 12),
11. (10, 3, 'PT', 'B', 12),
12. (11, 1, 'CC', 'B', 8),
13. (12, 1, 'CC', 'B', 7),
14. (13, 1, 'CC', 'B', 7),
15. (14, 1, 'HS', 'B', 5),
16. (15, 1, 'HS', 'B', 5),
17. (16, 2, 'PT', 'B', 12),
18. (17, 2, 'PT', 'B', 12),
19. (18, 3, 'PT', 'B', 12),
20. (19, 3, 'PT', 'B', 12),
21. (20, 3, 'PT', 'B', 12);
```

2.19) Insert into table thuoc

```
1. INSERT INTO `thuoc` (`MaThuoc`, `TenThuoc`, `TacDung`, `Gia`, `HSD`) VALUES
2. (1, 'Molnupiravir', 'Ức chế quá trình sao chép của virus SARS-CoV-2', 1500,
'2024-12-31'),
3. (2, 'Paxlovid', 'Ức chế quá trình sao chép của virus SARS-CoV-2', 2000,
'2025-01-01'),
4. (3, 'Ritonavir', 'Ức chế enzyme protease của virus SARS-CoV-2', 500, '2025-
02-01'),
5. (4, 'Verocell', 'Ức chế SARS-CoV-2', 100, '2024-02-01'),
6. (5, 'Baricitinib', 'Ức chế enzyme JAK1 và JAK2, giúp giảm viêm', 1000, '2025-
03-01'),
7. (6, 'Tocilizumab', 'Ức chế enzyme IL-6, giúp giảm viêm', 2000, '2025-04-01'),
8. (7, 'Dexamethason', 'Thuốc chống viêm corticosteroid', 500, '2025-05-01'),
9. (8, 'Methylprednisolon', 'Thuốc chống viêm corticosteroid', 1000, '2025-06-
01'),
10. (9, 'Prednisolon', 'Thuốc chống viêm corticosteroid', 1500, '2025-07-01'),
11. (10, 'Oxygen', 'Giúp cung cấp oxy cho cơ thể', 100, '2025-08-01'),
12. (11, 'Paracetamol', 'Hạ sốt, giảm đau', 12, '2024-12-31'),
13. (12, 'Bamlanivimab/etesevimab', 'Trung hòa virus SARS-CoV-2', 1000, '2032-12-
31'),
14. (13, 'Ibuprofen', 'Giảm đau, hạ sốt, chống viêm', 20, '2025-12-31'),
15. (14, 'Aspirin', 'Giảm đau, hạ sốt, chống viêm, chống đông máu', 30, '2026-12-
31'),
16. (15, 'Dexamethason', 'Chống viêm, chống dị ứng', 50, '2027-12-31'),
```



```

17. (16, 'Molnupiravir', 'Ngăn chặn sự nhân lên của virus SARS-CoV-2', 100,
'2028-12-31'),
18. (17, 'Nirmatrelvir/ritonavir', 'Ngăn chặn sự nhân lên của virus SARS-CoV-2',
200, '2029-12-31'),
19. (18, 'Molnupiravir/ritonavir', 'Ngăn chặn sự nhân lên của virus SARS-CoV-2',
300, '2030-12-31'),
20. (19, 'Sotrovimab', 'Trung hòa virus SARS-CoV-2', 500, '2031-12-31');

```

3) Alter Table

3.1) Alter Table BENHNHAN

```

1. ALTER TABLE `BENHNHAN` ADD FOREIGN KEY (`Ma_Staff`) REFERENCES `staff`
(`Ma_Staff`);
2. ALTER TABLE `BENHNHAN` ADD FOREIGN KEY (`Ma_Nurse`) REFERENCES `Nurse`
(`Ma_Nurse`);

```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng BENHNHAN:

+ ref: BENHNHAN.Ma_Staff > staff.Ma_Staff
+ ref: BENHNHAN.Ma_Nurse > Nurse.Ma_Nurse

3.2) Alter Table XetNghiem

```

1. ALTER TABLE `XetNghiem` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN`
(`MaBN`);

```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng XetNghiem:

+ ref: XetNghiem.MaBN > BENHNHAN.MaBN

3.3) Alter Table BenhLyDiKem

```

1. ALTER TABLE `BenhLyDiKem` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN`
(`MaBN`);

```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng BenhLyDiKem:

+ ref: BenhLyDiKem.MaBN > BENHNHAN.MaBN

3.4) Alter Table TrieuChungThuong

```

1. ALTER TABLE `TrieuChungThuong` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN`
(`MaBN`);

```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng TrieuChungThuong:

+ ref: TrieuChungThuong.MaBN > BENHNHAN.MaBN

3.5) Alter Table TrieuChungNghiemTrong

```
1. ALTER TABLE `TrieuChungNghiemTrong` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN` (`MaBN`);
```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng TrieuChungNghiemTrong:
+ ref: TrieuChungNghiemTrong.MaBN > BENHNHAN.MaBN

3.6) Alter Table Staff_PhanVao_Phong

```
1. ALTER TABLE `Staff_PhanVao_Phong` ADD FOREIGN KEY (`MaPhong`) REFERENCES `PHONG` (`MaPhong`);  
2. ALTER TABLE `Staff_PhanVao_Phong` ADD FOREIGN KEY (`Ma_Staff`) REFERENCES `staff` (`Ma_Staff`);  
3. ALTER TABLE `Staff_PhanVao_Phong` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN` (`MaBN`);
```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng Staff_PhanVao_Phong:
+ ref: Staff_PhanVao_Phong.MaPhong > PHONG.MaPhong
+ ref: Staff_PhanVao_Phong.Ma_Staff > staff.Ma_Staff
+ ref: Staff_PhanVao_Phong.MaBN > BENHNHAN.MaBN

Ngoài việc thêm ràng buộc cho các table được giao thì em cũng thêm ràng buộc cho các table sau:

3.7) Alter Table staff, doctors, nurse, manager, volunteers

```
1. ALTER TABLE `staff` ADD FOREIGN KEY (`Ma_Staff`) REFERENCES `NhanVien` (`MaNV`);  
2. ALTER TABLE `Doctors` ADD FOREIGN KEY (`Ma_Doctors`) REFERENCES `NhanVien` (`MaNV`);  
3. ALTER TABLE `Nurse` ADD FOREIGN KEY (`Ma_Nurse`) REFERENCES `NhanVien` (`MaNV`);  
4. ALTER TABLE `Manager` ADD FOREIGN KEY (`Ma_Manager`) REFERENCES `NhanVien` (`MaNV`);  
5. ALTER TABLE `Manager` ADD FOREIGN KEY (`Ma_Doctors`) REFERENCES `Doctors` (`Ma_Doctors`);  
6. ALTER TABLE `Volunteers` ADD FOREIGN KEY (`Ma_Volunteers`) REFERENCES `NhanVien` (`MaNV`);
```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng staff, doctors, nurse:
+ ref: staff.Ma_Staff > NhanVien.MaNV
+ ref: Doctors.Ma_Doctors > NhanVien.MaNV
+ ref: Nurse.Ma_Nurse > NhanVien.MaNV
+ ref: Manager.Ma_Manager > NhanVien.MaNV
+ ref: Manager.Ma_Doctors > Doctors.Ma_Doctors

+ ref: Volunteers.Ma_Volunteers> NhanVien.MaNV

3.8) Alter Table Dieutri

```
1. ALTER TABLE `Dieutri` ADD FOREIGN KEY (`MaBN`) REFERENCES `BENHNHAN` (`MaBN`);
2. ALTER TABLE `Dieutri` ADD FOREIGN KEY (`Ma_Doctors`) REFERENCES `Doctors`
   (`Ma_Doctors`);
3. ALTER TABLE `Dieutri` ADD FOREIGN KEY (`MaThuoc`) REFERENCES `Thuoc`
   (`MaThuoc`);
```

Thêm ràng buộc khóa ngoại FOREIGN KEY cho bảng Dieutri:

+ ref: Dieutri.MaBN > BENHNHAN.MaBN

+ ref: Dieutri.Ma_Doctors > Doctors.Ma_Doctors

+ ref: Dieutri.MaThuoc > Thuoc.MaThuoc

VI/ TÀI LIỆU THAM KHẢO

Tài liệu môn CSDL nâng cao (Database Advance) năm học 2023-2024 của thầy Phan Trọng Nhân.

[1] Chapter_1 - Database System Concepts and Architecture.pdf

[2] Chapter_2-ERD.pdf

[3] Chapter_3 - EERD.pdf

[4] Chapter_4-RelationalDataModelAndRelationalMapping.pdf

[5] Chapter_6 - SQL.pdf

VII/ KẾT LUẬN

Sau khi hoàn tất công việc được giao thì em ra được những thành phẩm như: Nghiệp vụ của Quarantine camp, ERD, file SQL để import vào DMBS, Table Mapping, Datatype của từng cột. Sau đó gửi lên group nhóm để hoàn thành đồ án.

Qua quá trình thực hiện nhiệm vụ của mình, em có cơ hội học hỏi và rèn luyện nhiều kỹ năng mới cho bản thân, bao gồm:

+ Kỹ năng phân tích yêu cầu: kỹ năng này giúp em trong việc phân tích yêu cầu của khách hàng.

+ Kỹ năng thao tác với các công cụ liên quan đến database: giúp em có thể áp dụng trong tương lai khi đi làm việc.

+ Kỹ năng làm việc nhóm: một kỹ năng quan trọng và cần thiết đối với em.

+ Kỹ năng xử lý tình huống và fix lỗi: trong quá trình làm có những lỗi phát sinh, em có phản ứng ngay và sửa lỗi cho kịp tiến độ của mình.

Nhìn chung, qua đồ án lần này đã giúp em tích lũy được nhiều kinh nghiệm, cảm ơn Thầy Phan Trọng Nhân đã tạo điều kiện để em và nhóm hoàn thành đồ án.