Database Design of a Hospital Chain

Author: Diego Turri | **Date:** 19/07/2022 | **Skills:** Database design phases, SQL (DDL, DML and Queries), HTML, PHP, Javascript.

Index / Links

Scenario

Scope of work: The aim of this project is the creation of a simplified version for the management of a group of hospitals in a region.

Note 1: The application domain of interest is informally described below. It is underlined that, as generally happens in real cases, the specification may not be complete and may present contradictions and/or ambiguities. It is the designer's task to identify any critical points, analyze them, and propose design solutions congruent with the entire development of the project.

Note 2: Pay particular attention to the data that must be inserted into the database compared to those that can be calculated from them. Please remember that it is always possible to manage the data flow at the interface level adequately.

Scenario: Design a database for managing hospital companies (hospitals) in a region. Each hospital, identified by a code, name, and address, comprises departments (cardiology, orthopedics, general medicine, etc.) where nurses, administrative staff, and medical personnel work. Some, but not all, hospitals also offer an emergency room service. Each department is identified by its name and the hospital to which it belongs; for each department, a telephone number and visiting hours for the relatives of hospitalized patients are relevant. Some departments have operating rooms. The departments that provide for the hospitalization of patients include rooms, identified by their number, with a specified number of beds that can be occupied or empty. Among the medical personnel of each department, whose service length is of interest, are the figures of vice heads and heads; for the former, the date on which they assumed this role is of interest, while for the latter, the list of specializations obtained is of interest. It is important to note that each department has a single chief physician while it may have several vice heads. A vice chief physician can replace their chief physician if necessary for a specified period; keeping track of the replacement and the replacement period is essential. Each hospital can have laboratories/clinics for carrying out tests or an agreement with external laboratories/clinics. In the case of laboratories/clinics located within a hospital structure, the floor and room of the department in which it is located are relevant, while in the case of external laboratories/clinics, the address, telephone number, and opening hours to the public are of interest.

Tests are characterized by a unique code, a description, and the cost both in the public healthcare system and privately; in the case of specialist tests, it is essential to trace the doctor who prescribed them and memorize any warnings (for example, to come on an empty stomach, follow a specific diet, etc.). Each patient, identified by their Social Security Number and characterized by their personal data, can book the tests they need; for each booking, the following are relevant: booking date, date and time of the test, the laboratory/clinic where it will be performed, the cost regime, and the urgency according to a predefined scale (green, yellow, or red). Note that a patient can make multiple bookings for the same test over time but not on the same day. For each hospitalized patient, the date of hospitalization and any existing pathologies are relevant, while for discharged patients, the date of discharge is relevant. Note that a patient can be hospitalized multiple times for the same pathologies in the same department. For hospitals with emergency rooms, the shifts of the staff who work there are relevant, namely doctors and nurses belonging to the same hospital. Consider additional attributes that are useful/necessary to characterize the scenario in question.

Extracting insights from scenario description

Hospital:

- It is characterized by a code, a name, and an address.
- It is made up of departments.
- Some, but not all, offer an emergency room.
- It can either have its laboratories/clinics for carrying out tests or have an agreement with external laboratories/clinics.

Department:

- Examples of departments are cardiology, orthopedics, general medicine, etc…
- Nurses, administrative staff, and medical staff can work within a department.
- It is identified by name and the hospital to which it belongs.
- For each department, a telephone number and visiting hours for the relatives of hospitalized patients are relevant.
- Some departments have operating rooms, other hospitalizations, or even both.
- The departments for patients' hospitalization provide rooms to hospitalize them (beds?).
- Each department has a single chief physician, which can have several vice heads.
- A department is also characterized by the floor.
- A department has several rooms.

Room:

- It is identified by its number.
- Has a specified number of beds.
- It can contain an operating room, laboratory/clinic (also anything else? Yes).
- It could also have height, width, and length.

Bed:

- It can be free or occupied by patients.
- Medical staff (I think a subset of Staff seen at the end).
- Length of service affects.
- It is divided into vice primary, primary, and others.
- For vice heads, the date they assumed the role is of interest.

- For primary doctors, the list of specializations obtained.
- A vice chief physician, if needed, can replace their chief physician (so are they linked to a chief physician?) for a defined period.
- It is essential to keep track of the replacement and the period of replacement.
- Prescribes tests to a patient, possibly placing a warning from the doctor ("this patient needs ..., has a problem with ..., and so on...).

Laboratory/clinic:

- They are intended to be the same single concept.
- Performs tests on patients.
- It can be either internal or external to the hospital.
- Of the internal ones, the room in which it is located affects.
- For external ones, the address, telephone number, and opening hours to the public are relevant.

Operating room:

- Performs operations on patients.
- Provides several beds.
- It affects the room in which it is located.
- So, a laboratory/clinic can be put in one room. Should this also have beds?
- So, is a recovery room nothing more than an operating room or similar placed inside a room?

Examination:

- It is characterized by a unique code, a description, and the cost both in healthcare and privately.
- It is divided into specialist and non-specialist.
- It is crucial to trace the doctor who prescribed it.
- It can contain notes such as warnings.

Patient:

- Is identified by their Social Security Number.
- It is characterized by personal data.
- Book a test they need (you can book only tests that your doctor prescribes).
- Can make multiple bookings for the same exam over time but not on the same day.

- The patient is divided into hospitalized and discharged.
- Of those hospitalized, the date and any pathologies present are of interest.
- Of those discharged, the date of discharge is relevant.
- A patient can be hospitalized several times for the same pathologies in the same department.

Reservation:

- Relevant information includes the booking date, date and time of the exam, laboratory/clinic where it will be performed, cost regime (with Social Security Number or private), and urgency according to a predefined scale (green, yellow, or red). Is red the most urgent? yes

Emergency room:

- The shifts of the staff who work there are relevant.
- The staff who serve there must be doctors or nurses and must belong to the same hospital.

Staff:

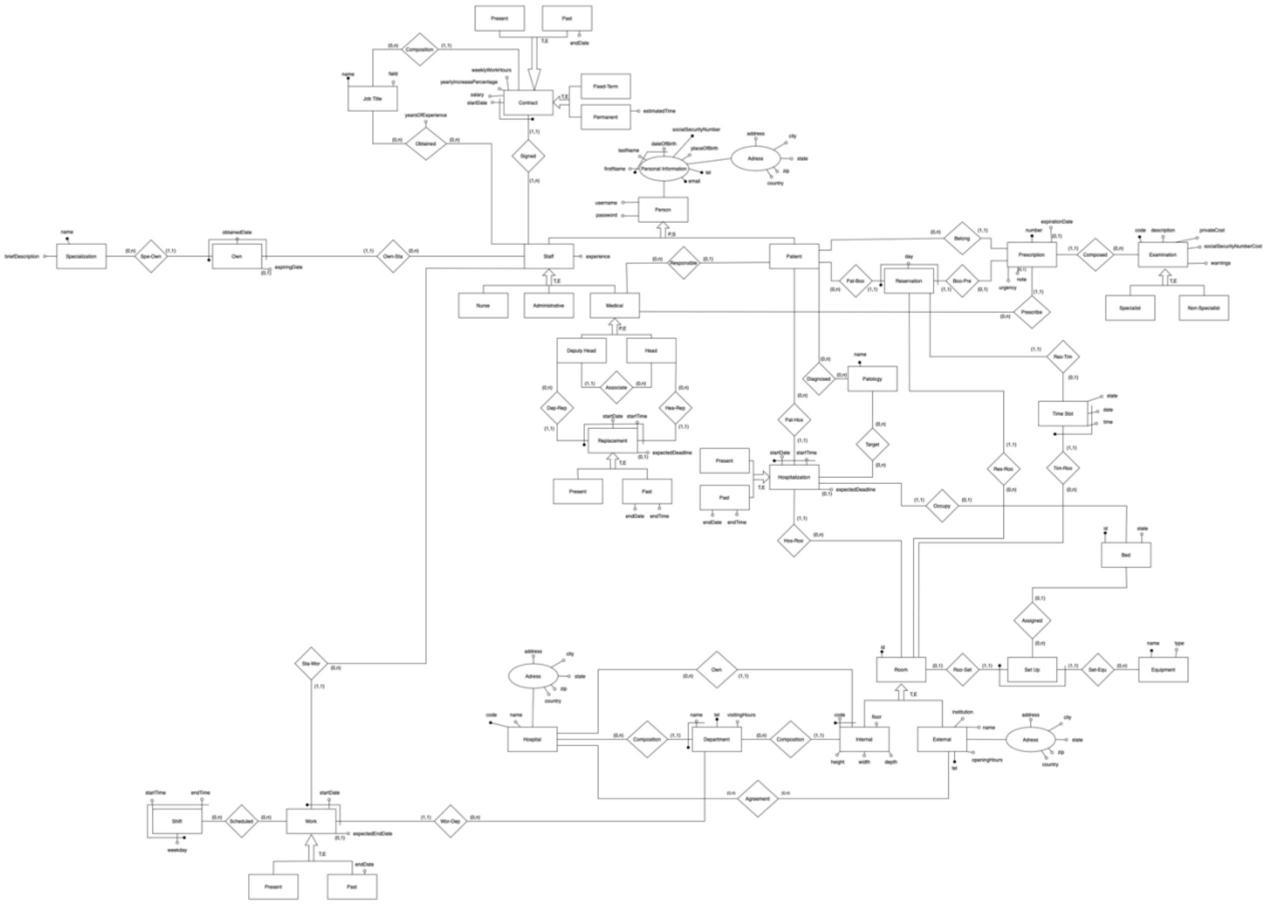
- It can be administrative staff, nurses, or medical staff.
- An example of medical personnel is vice head and head.
- The staff works in a department.
- For the medical personnel, the length of service and specializations obtained are of interest.
- A vice chief physician can replace the chief physician if necessary for a specified period.

Setup:

- A setup sets up a room.
- It is characterized by beds (not mandatory) and type of setup (operating room, laboratory/clinic).
- If the setup is a shelter, then beds are required, but if something else needs them, then beds are mandatory.

Conceptual design

2.1 - E-R schema



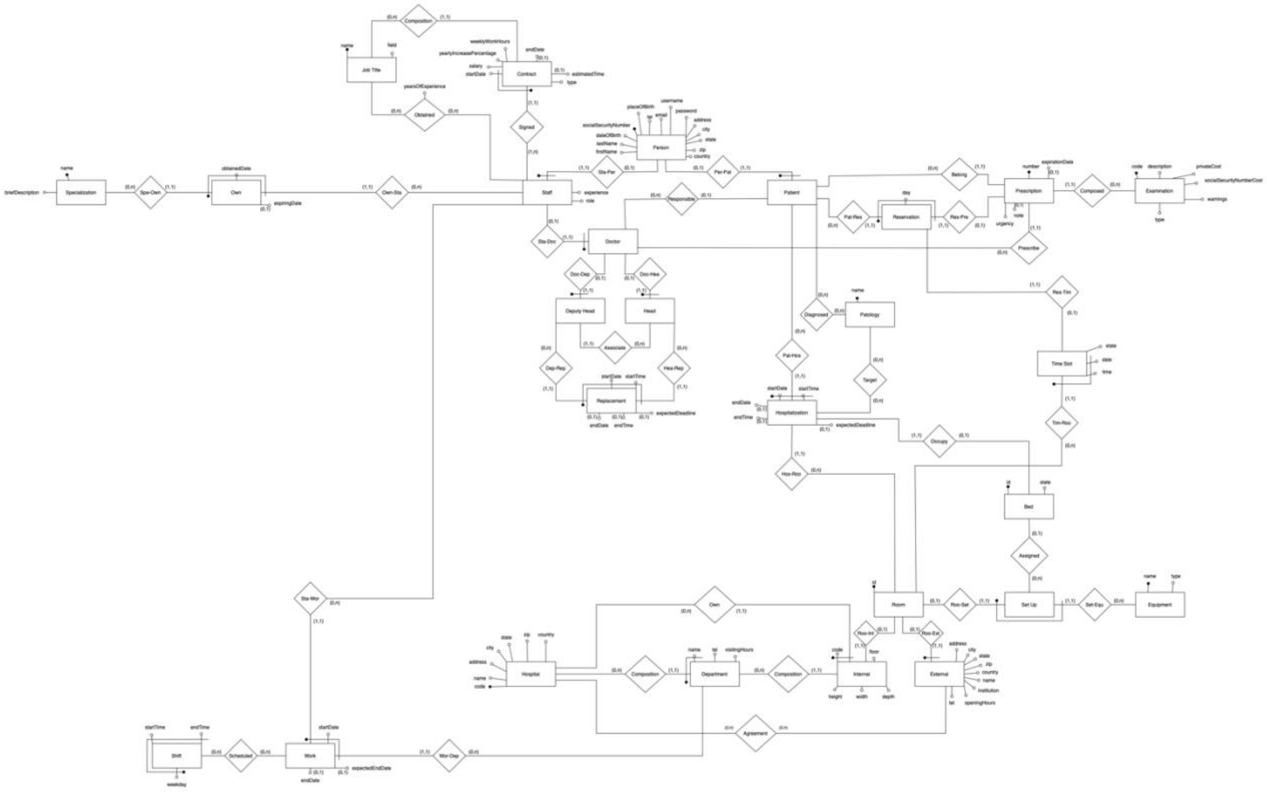
2.2 Domain constraints

- Only a relative of a hospitalized patient can visit it.
- All internal rooms belonging to the same department must belong to the same hospital.
- A bed can only be occupied if it is assigned.
- Some setups make the assignment of seats mandatory (e.g., hospitalization)
- A vice chief physician can only replace their chief physician and cannot replace more than one at a time.

- A doctor can only prescribe tests for a patient for whom they are responsible.
- A patient is only allowed to schedule exams that are prescribed them.
- A patient cannot be hospitalized if he is already being hospitalized (hospitalization has not finished “end date” and “end time” yet).
- Each department can have a maximum of one chief physician.
- A hospitalization must be conducted in a specially set room.
- An examination must be conducted in a specially set room (laboratory/clinic).
- An employee must have finished working to start a new job in the same department (end date before start date).
- If a department is an emergency room, then only doctors or nurses can work there.
- Weekly day can only have 7 values, the days of the week.
- An employee can work in multiple departments but only in the same hospital.
- An employee cannot have an employment contract if there is already one in progress (start date, end date).
- Job (start date, end date) must only be assigned to an employee with a current contract in that period (the contract cannot expire before the job's end date).
- A vice chief physician can replace their chief physician only during their work shift.
- Only a vice chief physician can replace the chief physician, not vice versa.
- A vice chief physician can replace his chief physician only if he is not already doing so (the end date must have passed).
- A patient cannot be hospitalized for pathologies that they don’t have.
- A hospitalization must occupy an unoccupied bed space.
- If the type of setup is a hospitalization room, then the beds are mandatory.

Logical design

4.1 - Restructured E-R schema



4.2 - Added domain constraints

- A room can be either internal or external.
- If an employee's role is "Doctor," then the association is mandatory; if it is not, then that association is prohibited.
- An employee can have one of three roles: Nurse, Administrative, or Doctor.
- In Replacement and Hospitalization, if endTime is present, endDate must also be present and vice versa.
- Contract (expectedEndDate) is mandatory if Contract(type) is a "Fixed-Term".

Orange indicates other possible unique identifiers

Purple indicates optional values

JobTitle(name, field)

Specialization(name, briefDescription)

Person(socialSecurityNumber, firstName, lastName, dateOfBirth, placeOfBirth, tel, email, address, city, state, zip, country, username, password)

Patient(socialSecurityNumber, doctor)

Prescription(number, expirationDate, urgency, note, doctor, examination, patient)

Examination(code, description, privateCost, socialSecurityNumberCost, warnings, type)

Patology(name)

Bed(id, state, room, equipment)

Equipment(name, type)

Room(id)

ExternalRoom(id, openingHours, tel, address, city, state, zip, country)

Hospital(code, name, address, city, state, zip, country)

Shift(weekday, startTime, endTime)

Employee(socialSecurityNumber, experience, role)

Contract(employee, startDate, salary, yearlyIncreasePercentage, weeklyWorkHours, expectedEndDate, type, JobTitle, endDate)

Own(employee, Specialization, obtainedDate, expiringDate)

Doctor(socialSecurityNumber)

ViceChiefPhysician(socialSecurityNumber, chiefPhysician)

ChiefPhysician(socialSecurityNumber)

Replacement(viceChiefPhysician, chiefPhysician, startDate, startTime, expectedEndDate, endDate, endTime)

Reservation(Prescription, patient, day, timeSlotDate, timeSlotTime, timeSlotRoom)

Hospitalization(patient, startDate, startTime, expectedEndDate, room, bed, endDate, endTime)

SetUp(equipment, room)

Internal(code, department, floor, height, width, depth, roomId)

Department(id, hospital, name, tel, visitingHours)

Work(employee, department, startDate, expectedEndDate, endDate)

Obtained(employee, JobTitle, yearsOfExperience)

Diagnosed(patient, patology)

Target(hospitalizedPatient, hospitalizedStartDate, hospitalizedStartTime, patology)

Agreement(hospital, externalRoom)

Scheduled(employee, department, startDate, shiftWeekday, shiftStartTime, shiftEndTime)

TimeSlot(date, time, room, state)

Physical design (SQL)

5.1 Data Definition Language (DDL)

```
CREATE TABLE JobTitle(
name varchar(50) PRIMARY KEY,
field varchar(50) NOT NULL
);

CREATE TABLE Specialization(
name varchar(50) PRIMARY KEY,
briefDescription varchar(500) NOT NULL
);

CREATE TABLE Person(
socialSecurityNumber char(11) PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
dateOfBirth date NOT NULL CHECK(dateOfBirth > '1900-01-01'),
placeOfBirth varchar(30) NOT NULL,
tel decimal(10,0) NOT NULL UNIQUE,
email varchar(50) NOT NULL UNIQUE,
address varchar(50) NOT NULL,
city varchar(30) NOT NULL,
state varchar(50) NOT NULL,
zip decimal(5,0) NOT NULL,
country varchar(30) NOT NULL,
username varchar(30) DEFAULT 'Guest',
password varchar(30) DEFAULT 'Guest',
UNIQUE(username, password),
UNIQUE(firstName, lastName, dateOfBirth)
);

CREATE TABLE Patient(
socialSecurityNumber char(11) PRIMARY KEY,
doctor char(11)
);

CREATE TABLE Prescription(
number varchar(20) PRIMARY KEY,
expirationDate date CHECK(expirationDate >= CURRENT_TIMESTAMP),
urgency decimal(1,0) NOT NULL,
note varchar(50),
doctor char(11) NOT NULL,
examination varchar(20) NOT NULL,
patient char(11) NOT NULL
);

CREATE TABLE Examination(
code varchar(20) PRIMARY KEY,
description varchar(300) NOT NULL,
privateCost decimal(7,2) NOT NULL,
socialSecurityNumberCost decimal(7,2) NOT NULL,
warnings varchar(1000) NOT NULL DEFAULT 'none',
type varchar(300) NOT NULL
);

CREATE TABLE Patology(
name varchar(50) PRIMARY KEY
);

CREATE TABLE Bed(
id varchar(6) PRIMARY KEY,
state varchar(8) NOT NULL,
room varchar(6),
equipment varchar(100)
);
```

```
CREATE TABLE Equipment(

name varchar(100) PRIMARY KEY,

type varchar(50) NOT NULL

);


CREATE TABLE Room(

id varchar(6) PRIMARY KEY

);


CREATE TABLE ExternalRoom(

id varchar(6) PRIMARY KEY,

openingHours varchar(100) NOT NULL,

tel decimal(10,0) NOT NULL UNIQUE,

address varchar(50) NOT NULL,

city varchar(30) NOT NULL,

state varchar(50) NOT NULL,

zip decimal(5,0) NOT NULL,

country varchar(30) NOT NULL

);


CREATE TABLE Hospital(
code varchar(3) PRIMARY KEY,
name varchar(100),
address varchar(50) NOT NULL,
city varchar(30) NOT NULL,
state varchar(50) NOT NULL,
zip decimal(5,0) NOT NULL,
country varchar(30) NOT NULL
);

CREATE TABLE Shift(
weekday varchar(9),
startTime varchar(10),
endTime varchar(10),
PRIMARY KEY(weekday, startTime, endTime)
);

CREATE TABLE Employee(
socialSecurityNumber char(11) PRIMARY KEY,
experience decimal(2,0) NOT NULL DEFAULT 0,
role varchar(100) NOT NULL
);
```

```
CREATE TABLE Contract(
employee char(11),
startDate date CHECK(startDate>=CURRENT_TIMESTAMP),
salary decimal(8,2) NOT NULL,
yearlyIncreasePercentage decimal (4,2) NOT NULL,
weeklyWorkHours decimal(2,0) NOT NULL,
expectedEndDate date CHECK(expectedEndDate>=startDate),
type varchar(50) NOT NULL,
JobTitle varchar(100) NOT NULL,
endDate date CHECK(endDate>=startDate),
PRIMARY KEY(employee, startDate)
);

CREATE TABLE Own(
employee char(11),
Specialization varchar(100),
obtainedDate date,
expiringDate date,
PRIMARY KEY(employee, Specialization, obtainedDate)
);

CREATE TABLE Doctor(
```

```sql
socialSecurityNumber char(11) PRIMARY KEY
);

CREATE TABLE ViceChiefPhysician(
socialSecurityNumber char(11) PRIMARY KEY,
chiefPhysician char(11) NOT NULL
);

CREATE TABLE ChiefPhysician(
socialSecurityNumber char(11) PRIMARY KEY
);

CREATE TABLE Replacement(
viceChiefPhysician char(11),
chiefPhysician char(11),
startDate date CHECK(startDate>=CURRENT_TIMESTAMP),
startTime varchar(8),
expectedEndDate date CHECK(expectedEndDate>=startDate),
endDate date CHECK(endDate>=startDate),
endTime varchar(8),
PRIMARY KEY(viceChiefPhysician, chiefPhysician, startDate, startTime)
);
```

```
CREATE TABLE Reservation(

Prescription varchar(20),

patient char(11),

day date,

timeSlotDate date NOT NULL CHECK(timeSlotDate>=day),

timeSlotTime varchar(8) NOT NULL,

timeSlotRoom varchar(6) NOT NULL,

PRIMARY KEY(Prescription, patient, day)

);


CREATE TABLE Hospitalization(

patient char(11),

startDate date CHECK(startDate>=CURRENT_TIMESTAMP),

startTime varchar(8),

expectedEndDate date CHECK(expectedEndDate>=startDate),

room varchar(6) NOT NULL,

bed varchar(6) NOT NULL,

endDate date CHECK(endDate>=startDate),

endTime varchar(8),

PRIMARY KEY(patient, startDate, startTime)

);
```

```
CREATE TABLE SetUp(
equipment varchar(100),
room varchar(6),
PRIMARY KEY(equipment, room)
);

CREATE TABLE Internal(
code varchar(6),
department decimal(5,0),
floor decimal(2,0) NOT NULL,
height decimal(8,2) NOT NULL,
width decimal(8,2) NOT NULL,
depth decimal(8,2) NOT NULL,
roomId varchar(6) NOT NULL UNIQUE,
PRIMARY KEY(code, department)
);

CREATE TABLE Department(
id decimal(5,0) PRIMARY KEY,
hospital varchar(3),
name varchar(100),
tel decimal(10,0) NOT NULL UNIQUE,
visitingHours varchar(100) NOT NULL
);

CREATE TABLE Work(
employee char(11),
department decimal(5,0),
startDate date CHECK(startDate>=CURRENT_TIMESTAMP),
expectedEndDate date CHECK(expectedEndDate>=startDate),
endDate date CHECK(endDate>=startDate),
PRIMARY KEY(employee, department, startDate)
);

CREATE TABLE Obtained(
employee char(11),
JobTitle varchar(50),
yearsOfExperience decimal(2,0) NOT NULL,
PRIMARY KEY(employee, JobTitle)
);

CREATE TABLE Diagnosed(
patient char(11),
patology varchar(50),
PRIMARY KEY(patient, patology)
);

CREATE TABLE Target(
hospitalizedPatient char(11),
hospitalizedStartDate date,
hospitalizedStartTime varchar(8),
patology varchar(50) NOT NULL,
PRIMARY KEY(hospitalizedPatient, hospitalizedStartDate, hospitalizedStartTime)
);

CREATE TABLE Agreement(
hospital varchar(3),
externalRoom varchar(6),
PRIMARY KEY(hospital, externalRoom)
);

CREATE TABLE Scheduled(
employee char(11),
department decimal(5,0),
startDate date,
```

```sql
shiftWeekday varchar(9),
shiftStartTime varchar(8),
shiftEndTime varchar(8),
PRIMARY KEY(employee, department, startDate, shiftWeekday, shiftStartTime, shiftEndTime)
);

CREATE TABLE TimeSlot(
date date NOT NULL,
time varchar(11) NOT NULL,
room varchar(6) NOT NULL,
state varchar(1) NOT NULL,
PRIMARY KEY(date, time, room)
);



ALTER TABLE patient ADD FOREIGN KEY (socialSecurityNumber) REFERENCES
Person(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE patient ADD FOREIGN KEY (doctor) REFERENCES
doctor(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE Prescription ADD FOREIGN KEY (doctor) REFERENCES
doctor(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE Prescription ADD FOREIGN KEY (examination) REFERENCES
examination(code) ON UPDATE CASCADE;
```

```
ALTER TABLE Prescription ADD FOREIGN KEY (patient) REFERENCES patient(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE bed ADD FOREIGN KEY (room, equipment) REFERENCES setUp(room, equipment) ON UPDATE CASCADE;

ALTER TABLE externalRoom ADD FOREIGN KEY (id) REFERENCES room(id) ON UPDATE CASCADE;

ALTER TABLE employee ADD FOREIGN KEY (socialSecurityNumber) REFERENCES Person(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE contract ADD FOREIGN KEY (employee) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE contract ADD FOREIGN KEY (JobTitle) REFERENCES JobTitle(name) ON UPDATE CASCADE;

ALTER TABLE own ADD FOREIGN KEY (employee) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE own ADD FOREIGN KEY (Specialization) REFERENCES Specialization(name) ON UPDATE CASCADE;

ALTER TABLE doctor ADD FOREIGN KEY (socialSecurityNumber) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE viceChiefPhysician ADD FOREIGN KEY (socialSecurityNumber) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE viceChiefPhysician ADD FOREIGN KEY (chiefPhysician) REFERENCES chiefPhysician(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE chiefPhysician ADD FOREIGN KEY (socialSecurityNumber) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE replacement ADD FOREIGN KEY (viceChiefPhysician) REFERENCES viceChiefPhysician(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE replacement ADD FOREIGN KEY (chiefPhysician) REFERENCES chiefPhysician(socialSecurityNumber) ON UPDATE CASCADE;
ALTER TABLE reservation ADD FOREIGN KEY (Prescription) REFERENCES Prescription(number) ON UPDATE NO ACTION;

ALTER TABLE reservation ADD FOREIGN KEY (patient) REFERENCES patient(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE reservation ADD FOREIGN KEY (timeSlotDate, timeSlotTime, timeSlotRoom) REFERENCES timeSlot(date, time, room) ON UPDATE CASCADE;

ALTER TABLE hospitalization ADD FOREIGN KEY (patient) REFERENCES patient(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE hospitalization ADD FOREIGN KEY (room) REFERENCES room(id) ON UPDATE CASCADE;

ALTER TABLE hospitalization ADD FOREIGN KEY (bed) REFERENCES bed(id) ON UPDATE CASCADE;

ALTER TABLE setUp ADD FOREIGN KEY (equipment) REFERENCES equipment(name) ON UPDATE CASCADE;

ALTER TABLE setUp ADD FOREIGN KEY (room) REFERENCES room(id) ON UPDATE CASCADE;

ALTER TABLE internal ADD FOREIGN KEY (department) REFERENCES department(id) ON UPDATE CASCADE;

ALTER TABLE internal ADD FOREIGN KEY (roomId) REFERENCES room(id) ON UPDATE CASCADE;

ALTER TABLE department ADD FOREIGN KEY (hospital) REFERENCES hospital(code) ON UPDATE CASCADE;

ALTER TABLE work ADD FOREIGN KEY (employee) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE work ADD FOREIGN KEY (department) REFERENCES department(id) ON UPDATE CASCADE;

ALTER TABLE obtained ADD FOREIGN KEY (employee) REFERENCES employee(socialSecurityNumber) ON UPDATE CASCADE;
ALTER TABLE obtained ADD FOREIGN KEY (JobTitle) REFERENCES JobTitle(name) ON
UPDATE CASCADE;

ALTER TABLE diagnosed ADD FOREIGN KEY (patient) REFERENCES
patient(socialSecurityNumber) ON UPDATE CASCADE;

ALTER TABLE diagnosed ADD FOREIGN KEY (patology) REFERENCES patology(name) ON
UPDATE CASCADE;

ALTER TABLE target ADD FOREIGN KEY (patology) REFERENCES patology(name) ON
UPDATE CASCADE;

ALTER TABLE target ADD FOREIGN KEY (hospitalizedPatient, hospitalizedStartDate,
hospitalizedStartTime) REFERENCES hospitalization(patient, startDate, startTime) ON UPDATE
CASCADE;

ALTER TABLE agreement ADD FOREIGN KEY (hospital) REFERENCES hospital(code) ON
UPDATE CASCADE;

ALTER TABLE agreement ADD FOREIGN KEY (externalRoom) REFERENCES
externalRoom(id) ON UPDATE CASCADE;

ALTER TABLE scheduled ADD FOREIGN KEY (employee, department, startDate)
REFERENCES work(employee, department, startDate) ON UPDATE CASCADE;

ALTER TABLE scheduled ADD FOREIGN KEY (shiftWeekday, shiftStartTime, shiftEndTime)
REFERENCES shift(weekday, startTime, endTime) ON UPDATE CASCADE;

ALTER TABLE timeSlot ADD FOREIGN KEY (room) REFERENCES room(id) ON UPDATE
CASCADE;
```

Web application

https://github.com/diegoturri/Hospitals-Database-Design-Project