DATA DESIGN PART II

Colin Steele and Raul Ramirez

4/21/22

INTRODUCTION:

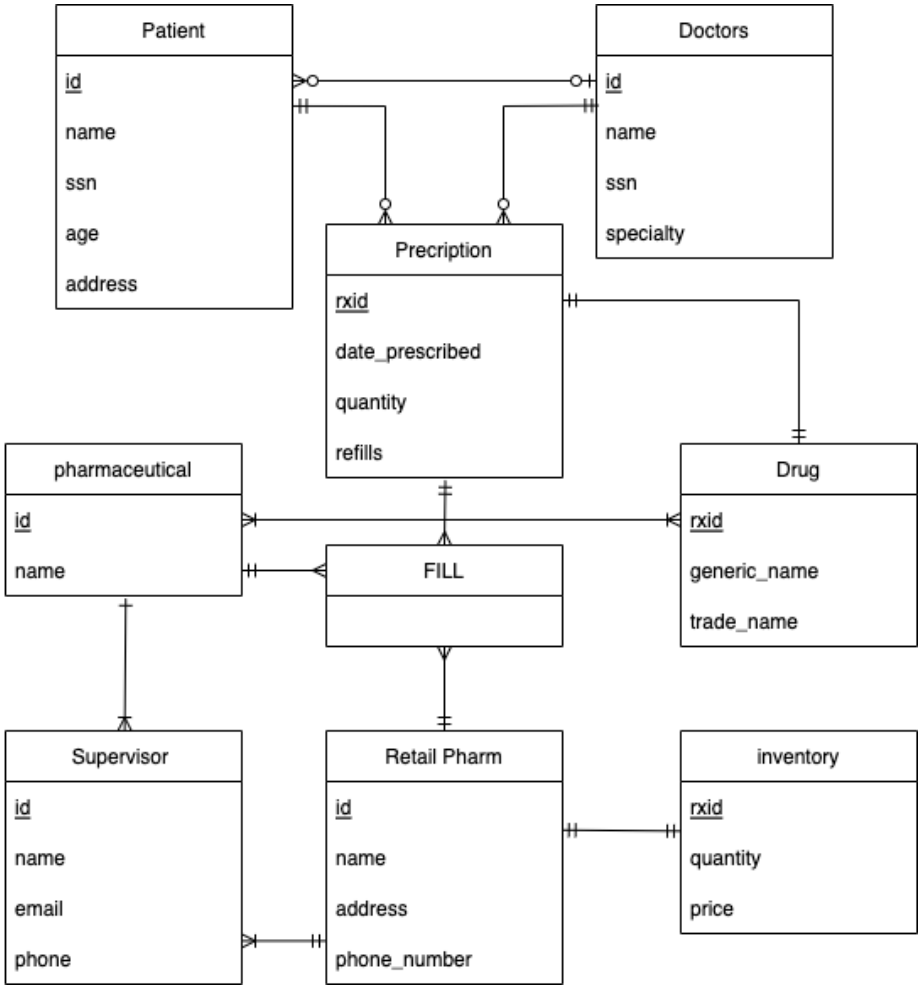
The main purpose of this project is to make a database to keep track of the history and current prescription drugs of a patient. In doing so we need to keep track of:

- Patient Personal Information
- Doctors who can prescribe drug
- Drug information
- Prescription information
- Pharmaceutical manufacturer
- Pharmacy locations
- The existence of the active contracts
- The supervisors for the Pharmaceutical companies and Pharmacies

Furthermore we will also be keeping track of the prices at every location. Since the price and availability may vary between pharmaceutical locations, we will also need to keep track of the contracts between a manufacturer and store.

The only major assumption being that stock would be part of the consideration. For the fact that a drug can be prescribed, but if there is no availability then at a location or chain then it would not be wise to direct a patient to that location. Even if it would have been cheaper or closer, medication can be vital to patients survival or comfort of living.

ER MODEL:



Each container in this diagram represents an entity/table in the database. The lines connecting the tables represent the relation between two tables and then the end points elaborate on that relation. For example Looking at the Patients and Doctors diagram we can see that there exists a relation between them. A patient's relation to doctors is a one to many relation as patients must have at least one primary doctor, but can visit many doctors. This is indicated by the far right end point. Then a doctor's relation to patents is zero to many, as a doctor may have zero patients or many at any given time. And this is indicated by the far left end point. Similarly the diagram shows the rest of the relation between each entity. A table not shown here one in which keeps track of a list of doctors and their patients. Rather than repeat doctor information and or patient information a seperate table will be used to keep track of doctor ids and patient ids.

RELATIONAL SCHEMA:

```
create table patient(
    id int primary key,
    name varchar(25) not null,
    address varchar(100) not null,
    ssn char(11) not null,
    dob char(10) not null,
    doc_id int not null,
    foreign key (doc_id) references
doctor(id)
);

create table prescription(
    rxid numeric(5,0),
    patient_id int not null,
    doc_id int not null,
    qty numeric(5,3) not null,
    refills int not null,
    date_pr char(10) not null,
    primary key(rxid,patient_id),
    foreign key(rxid) references drug(rxid),
    foreign key(patient_id) references patient(id),
    foreign key(doc_id) references doctor(id)
);

    create table doctor(
        id int primary key,
        name varchar(25),
        ssn  char(11),
        specialty varchar(20)
    );

create table fill(
    rxid int not null,
    pharma int not null,
    retail int not null,
    qty numeric(6,3) not null,
    patient int,
    fill_date char(10) not null,
    foreign key(rxid) references drug(rxid),
    foreign key(pharma) references
pharmaceutical,
    foreign key(patient) references patient(id)
);

create table drug(
    rxid numeric(5,0) check (rxid > 10000),
    generic_name varchar(50) not null,
    trade_name varchar(50),
    primary key(rxid)
);

create table supervisor(
    id int primary key,
    name varchar (25) not null,
    email varchar(45) not null,
    phone varchar (15)
);
create table pharmaceutical(
    id int primary key,
    name varchar(30) not null,
    sup_id int not null,
    foreign key (sup_id) references supervisor(id)
);

create table retail_pharma(
    id int primary key,
    name varchar(30) not null,
    address varchar(100) not null,
    phone varchar(15) not null
);

create table inventory(
    rxid int not null,
    retail_id int not null,
    qty int not null,
    price numeric (5,2),
    primary key(rxid,retail_id),
    foreign key(rxid) references drug(rxid),
    foreign key(retail_id) references retail_pharma(id)
);
```

EXAMPLE SQL QUIERES:

What prescriptions were prescribed by Dr Paulillo (id 4) so far in 2022 ?

```
select * from PRESCRIPTION
where DOC_ID = 4
and right(DATE_PR,4) = 2022
```

Which retail pharmacy sells the drug, with rxid = 10278, the cheapest ?

```
select top 1 retail_id from INVENTORY
where rxid = 10278
order by price asc;
```

Who prescribed Jerrold Aubry his artificial tears prescription and where did he get it filled ?

```
select dr.name,rp.name from PRESCRIPTION pr natural join DRUG
join fill f on (pr.rxid,pr.patient_id) = (f.rxid,f.patient)
join RETAIL_PHARMA rp on f.retail = rp.id
join DOCTOR dr on pr.doc_id = dr.id
where patient_id in ( select p.id from PATIENT p where NAME = 'Jerrold Aubry' );
```

What are the names and addresses for patients who have unfilled prescriptions?

```
select p.name, p.address from PATIENT as p
join PRESCRIPTION as pr on p.id = pr.patient_id where not exists(select fill.rxid,
fill.patient from fill where pr.rxid = fill.rxid and pr.patient_id = fill.patient)
```

What pharmacies have ‘alitretinoin’ in stock?

```
select distinct * from RETAIL_PHARMA as rp
join INVENTORY as i on rp.id = i.retail_id
join DRUG as d on d.rxid = i.rxid where
(d.generic_name = 'alitretinoin' OR d.trade_name = 'alitretinoin') and i.qty > 0;
```

CONCLUSION:

What we ended up with was a working database where a pharmaceutical company can keep track of the movement and pricing of its drugs. The database manages to keep track of drug prescriptions, the doctor(s) who prescribed it and to whom, as well as where a patient gets their prescription filled, where a retail pharmaceutical gets their inventory and how they price it. With robust enough information that aid doctors, pharmacies, and pharmaceutical companies know when prescriptions are unfilled. That can be beneficial for multiple metrics.

This project has given us a better understanding as to the process involved in building a database. It showed us that the process involves a lot of communication between the client and design team as the requirements may not alway be thoroughly understood right from the beginning. Especially since implications of a relation or method in which to keep track of entities may not already have one onset of a project but can always be created with enough thought and consideration. Overall,I believe that we managed to provide 95% of what we believed the project required. Since the data was randomly generated, this resulted in prescription data not lining up with fill data . But we believe that this would be prevented with a software application layer.