Relational Database Design for Drug Store Chain

PREPARED BY

Eddie Sanchez

Yvonne Cruz

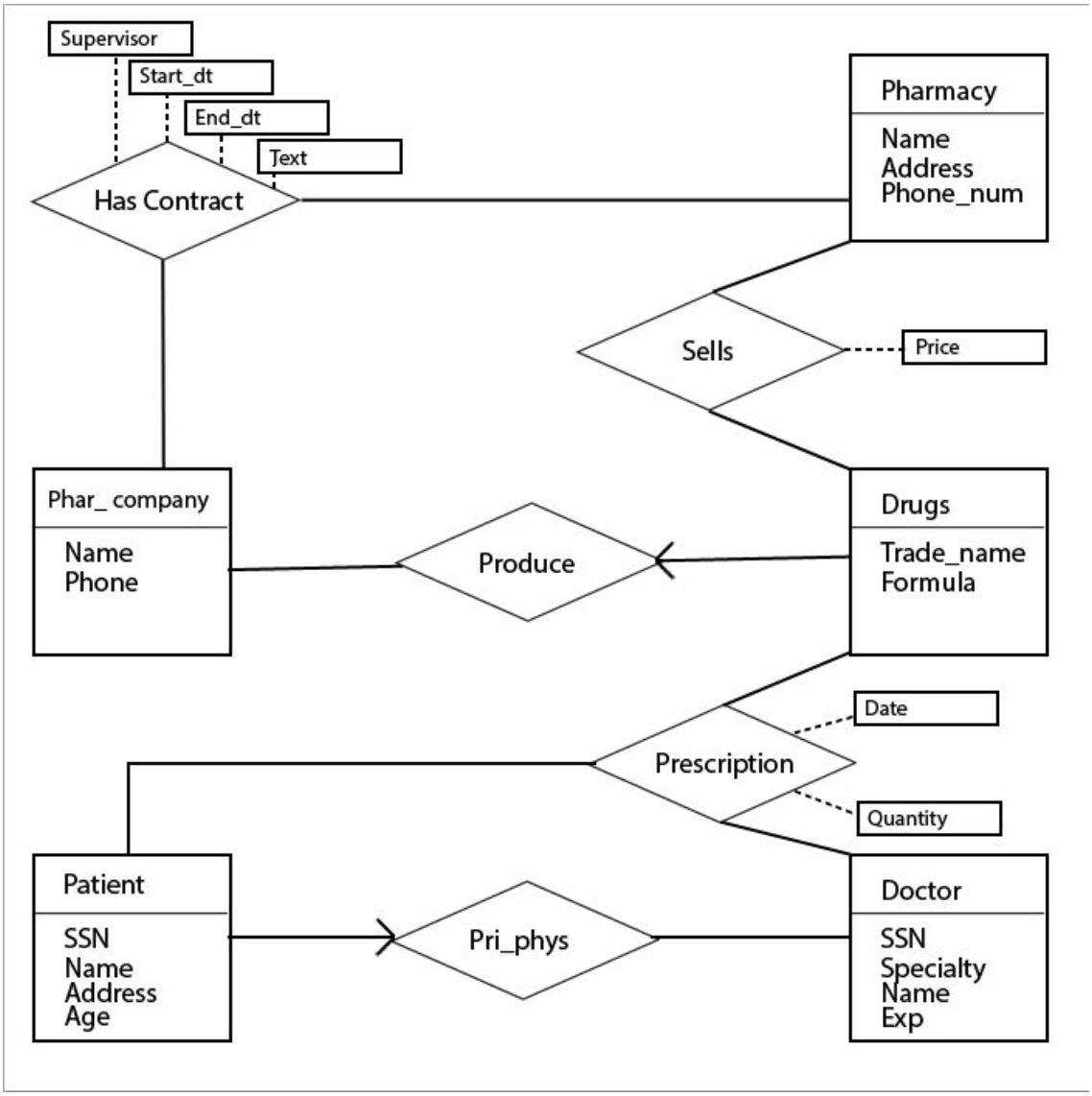
Contents

1.Introduction

This project deals with the management of a drug store chain. The chain is used to dispense prescription medications to patients. The challenge within a drug store chain is that all data has to be correctly organized so that there is no slow information processing and no loss of medical records. The system introduced will be very well organized with appropriate attributes needed to make it successful.

The aim is to provide a database system with the use of computer software and to fulfill the requirement to store important data while making it easily accessible and manipulative. This project will use Entity-Relationship modeling. An Entity-Relationship model is a high-level design of the database that shows database entities, attributes, and the relationships between entities. Entities are objects in the system that will be modeled like patients, doctors, and drugs. Attributes are elements that give the entities its characteristics like name, phone number, and address.

2. ER Model



This is the Entity Relationship design model for the database of the drug store chain. Creating the ER model was difficult because we were not that familiar with the tools available to create one. After deciding on a design software, implementing entities with its attributes and assigning their appropriate relationships with each other was easy to complete and transfer them onto this document.

The ER model gives a visual layout of the entities and attributes as well as where the links between entities are. Entities were divided accordingly so there will be little to no margin of error.

- Patient: has a many-to-one relationship with doctor entity.
- Pharmaceutical Company (Phar_co): has a one-to-many relationship with drugs entity.
- Pharmacy: has a many-to-many relationship with drugs entity.

3. Relational Schema

```
- - info about patients
create table Patient(
SSN             VARCHAR(15),                -social security number of patient
Name            VARCHAR(20),                -name of patient
Address         VARCHAR(25),                -address of patient
Age             NUMERIC(3,0)              -age of patient
primary key (SSN));

- - info about doctors
create table Doctor(
SSN             VARCHAR(15),              -social security number of doctor
Specialty       VARCHAR(20),              -specialty of doctor
Name            VARCHAR(20),              -name of doctor
Exp       VARCHAR(20)             -number of years of experience of doctor
primary key (SSN));

- - info about pharmaceutical companies
create table Phar_company(
Name            VARCHAR(20),        -name of pharmaceutical company
Phone           VARCHAR(15)          -phone number of pharmaceutical company
Primary key(Name));

- - info about pharmacies
create table Pharmacy(
Name            VARCHAR(20),            -name of pharmacy
Address         VARCHAR(25),            -address of pharmacy
Phone_num   VARCHAR(20))              -phone number of pharmacy
primary key(Name));

- - info about drugs
create table Drugs(
Trade_name  VARCHAR(20),          -trade name of drug
Formula         VARCHAR(20)             -formula of drug
Primary key(Trade_name));


- - info about prescription

Create table Prescription (
Pat_SSN         VARCHAR(15),          -social security of patient
Doc_SSN         VARCHAR(15),          -social security of doctor
Trade_name      VARCHAR(20),          -trade name of drug
Quantity        NUMERIC(10),          -quantity ordered of drug
Date            VARCHAR(8),           -date of prescription made
Primary key(Date),
```

```
Foreign key (Pat_SSN) references Patient on delete set null,
Foreign key (Doc_SSN) references Doctor on delete set null,
Foreign key (Trade_Name) references Drug on delete set null);

- - info about sells
create table (
Phar_Name          VARCHAR(5),                --name of pharmacy
Trade_Name         VARCHAR(20),               -trade name of drug
price              VARCHAr(20),               -price of drug
Primary key(Trade_Name),
Foreign key(Phar_Name) references Pharmacy on delete set null,
Foreign key(Trade_Name) references Drug on delete set null);


- - info about has contract
create table HasContract(
Phar_Co_Name       VARCHAR(20),           -name of pharmaceutical company
Phar_Name                VARCHAR(20),          -name of pharmacy
Supervisor         VARCHAR(20),           -name of supervisor
Start_dt           VARCHAR(8),            -start date of contract
End_dt             VARCHAR(8),            -end date of contract
Text               VARCHAR(20),           -info about contract
Primary key(start_dt),
Foreign key(Phar_Co_Name) references Phar_Company (name) delete set null,
Foreign key (Phar_name) references Pharmacy (name) on delete set null);
```

This is the relational schema that is derived from the ER model. It is divided into 8 tables: Patient (for patients), Doctor (for doctors), Phar_comany (for pharmaceutical companies), Pharmacy (for pharmacies), and Drugs (for prescription drugs), prescriptions, sells, and has contract. Each attribute in each table has been assigned a specific constraint in which only a certain amount of characters can be inserted into an attribute. There has also been assigned primary keys, which are unique keys that are not found in any other table.

4. Normalized Relational Schema

```
- - info about patients
create table Patient(
pat_ID          VARCHAR(5),              -patient identification number
SSN             VARCHAR(15),             -social security number of patient
Name            VARCHAR(20),             -name of patient
Address         VARCHAR(25),             -address of patient
Age             NUMERIC(3),              -age of patient
Doc_id          VARCHAR(5),              -doctor identification number
Phar_ID         VARCHAR(5),              -pharmacy identification number
primary key (ID),
Foreign key (Doc_ID) references Doctor on delete set null,
Foreign key (Phar_ID) references Pharmacy on delete set null);

- - info about doctors
create table Doctor(
Doc_ID        VARCHAR(5),              -doctor identification number
SSN           VARCHAR(15),             -social security number of doctor
Specialty     VARCHAR(20),             -specialty of doctor
Name          VARCHAR(20),             -name of doctor
Exp           NUMERIC(3),              -number of years of experience of doctor
primary key (ID));

- - info about pharmaceutical companies
create table Phar_company(
Phar_co_ID   VARCHAR(5),    --pharmaceutical company identification number
Name              VARCHAR(20),     -name of pharmaceutical company
Phone             VARCHAR(15)    -phone number of pharmaceutical company
Primary key(Phar_co_ID));

- - info about pharmacies
create table Pharmacy(
Phar_ID         VARCHAR(5),              --pharmacy identification number
Name            VARCHAR(20),              -name of pharmacy
Address         VARCHAR(25),             -address of pharmacy
Phone_num   VARCHAR(20),              -phone number of pharmacy
primary key(Phar_ID));

- - info about drugs
create table Drugs(
Drug_ID          VARCHAR(5),              --drug identification number
Trade_name    VARCHAR(20),              -trade name of drug
Formula          VARCHAR(20),             -formula of drug
Primary key(Drug_ID));
```

- - info about prescription

```
 Create table Prescription (
Presc_id        VARCHAR(5),      --prescription identification number
Pat_id          VARCHAR(5),      -- patient identification number
Doc_id          VARCHAR(5),      --doctor identification number
Drug_id         VARCHAR(5),      --drug identification number
Quantity        NUMERIC(3),      --quantity of drug
Date            VARCHAR(8),       --date when prescription was made
Primary key(Presc_id),
Foreign key (Pat_id) references Patient on delete set null,
Foreign key (Doctor_id) references Doctor on delete set null,
Foreign key (Drug_id) references Drug on delete set null);
```

- - info about sells

```
create table sells(
Trans_id       VARCHAR(5),        -- transaction identification number
Phar_ID          VARCHAR(5),      --pharmacy identification number
Drug_ID    VARCHAR(20),           --drug identification number
price          NUMERIC(4),        --price of drug
Primary key (trans_id),
Foreign key (phar_id) references pharmacy on delete set null,
Foreign key (drug_id) references drug on delete set null);
```

- - info about has contract

```
create table HasContract(
Contract_id             VARCHAR(5),     -contract identification number
Phar_Co_ID              VARCHAR(20),   -pharmaceutical company id
Phar_ID                 VARCHAR(20),    -pharmacy identification number
Supervisor              VARCHAR(20),     -name of supervisor
Start_dt                VARCHAR(8),      -start date of contract
End_dt                  VARCHAR(8),      -end date of contract
Text                    VARCHAR(20),      -info about contract
Primary key(contract_id),
Foreign key(Phar_Co_ID) references Phar_Company on delete set null,
Foreign key (Phar_ID) references Pharmacy on delete set null);
```

This is the normalized relational schema where certain fixes have been done in order to make the database a lot more successful. For example, instead of using social security numbers of patients and doctors as primary keys it was easier to assign identification numbers to the entities to make things flow a lot smoother. This resulted in better organization and better communication between entities when exchanging information.

5. Sample Data and SQL

Sample questions that can be answered with queries using the normalized data:

1. What is the average age of people who use Xanax?
2. What is the lowest amount of contracts a Pharmaceutical Company has?
3. What is the maximum amount of drugs one person has taken?
4. What drug yields the most money?
5. What is the average amount of patients a pharmacy has?

Knowing the averages/maximums/minimums of certain information will help better understand where things need to be and will help determine if something like a pharmacy is meeting its quota. Knowing the age of people who use a specific drug can help to know what the demographic is in selling that specific drug. Figuring out which Pharmaceutical Company has the least amount of contracts can help in knowing what company would be more susceptible in making a better contract with. Knowing the maximum amount of drugs that doctors prescribe can help with knowing how many drugs to order for each store. If a drug yields a good amount of revenue then better contracts and order supplies can be made. If a pharmacy has a lower than average patients then it is good to look into improving that store chain.

Sample Data:

```
INSERT INTO Patient VALUES('00001','987-65-4321','John Smith','123 Sesame St',41,'10000','20000');
INSERT INTO Patient VALUES('00002','789-56-1234','Ellen Smith','434 Concord St',18,'10001','20000');
INSERT INTO Patient VALUES('00003','354-65-4823','Jane Doe','555 Palm Ave',29,'10001','20003');
INSERT INTO Patient VALUES('00004','584-12-8945','Zack Santos','101 Doran Rd',56,'10002','20002');
INSERT INTO Patient VALUES('00005','848-45-4598','Elizabeth Cruz','644 Andrews Pl',25,'10004','20001');
INSERT INTO Patient VALUES('00006','590-65-4321','Kevin Nguyen','809 Lake St',39,'10003','20001');
INSERT INTO Patient VALUES('00007','333-22-0001','Ed Stevens','323 Seasice Ave',72,'10002','20000');

INSERT INTO Doctor VALUES('10000','200-22-2222','Internal Medicine','Dr. Alex Heald',16);
INSERT INTO Doctor VALUES('10001','263-77-1009','Family Medicine','Dr. Ronald Gonzales',6);
INSERT INTO Doctor VALUES('10002','444-22-2722','Internal Medicine','Dr. Jen Sanders',20);
INSERT INTO Doctor VALUES('10003','778-12-4444','Internal Medicine','Dr. Mike Li',9);
INSERT INTO Doctor VALUES('10004','908-24-6767','Emergency ','Dr. Angel Chu',5);

INSERT INTO Phar_company VALUES('30000','Johnson & Johnson','(818)-219-0909');
INSERT INTO Phar_company VALUES('30001','Roche','(800)-434-1111');
INSERT INTO Phar_company VALUES('30002','Pfizer','(800)-744-0030');
INSERT INTO Phar_company VALUES('30003','Novartis','(800)-333-3232');
INSERT INTO Phar_company VALUES('30004','Sanofi','(800)-200-1234');
INSERT INTO Phar_company VALUES('30005','Bayer','(800)-500-4321');

INSERT INTO Pharmacy VALUES('20000','CVS','345 Street St','(801)-909-7878');
INSERT INTO Pharmacy VALUES('20001','CVS','999 Avenue St','(801)-642-9090');
INSERT INTO Pharmacy VALUES('20002','Walgreens','999 Road St','(801)-909-6222');
INSERT INTO Pharmacy VALUES('20003','CVS','722 Circle St','(801)-671-0000');
INSERT INTO Pharmacy VALUES('20004','Rite Aid','587 Same St','(801)-343-1111');

INSERT INTO Drugs VALUES('40000','Xanax','alprazolam');
INSERT INTO Drugs VALUES('40001','Bleomycin','BLEOMYCIN SULFATE');
INSERT INTO Drugs VALUES('40002','Nifedipine','Nifedipine');
INSERT INTO Drugs VALUES('40003','AXERT', 'almotriptan malate');
INSERT INTO Drugs VALUES('40004','PRAX', 'pramoxine hydrochloride');
INSERT INTO Drugs VALUES('40005','Lithium Carbonate', 'Lithium Carbonate');
```

```
INSERT INTO Prescription VALUES('50000','00001','10000','40000',20,'12/30/18');
INSERT INTO Prescription VALUES('50001','00007','10003','40003',10,'1/30/19');
INSERT INTO Prescription VALUES('50002','00006','10002','40003',7,'5/30/19');
INSERT INTO Prescription VALUES('50003','00004','10002','40003',14,'3/09/19');
INSERT INTO Prescription VALUES('50004','00003','10002','40002',21,'7/11/19');
INSERT INTO Prescription VALUES('50005','00002','10001','40000',7,'12/12/19');
INSERT INTO Prescription VALUES('50006','00001','10001','40004',30,'5/5/19');
INSERT INTO Prescription VALUES('50007','00007','10003','40001',60,'2/28/19');

INSERT INTO Sells VALUES('60000','20000','40000',50);
INSERT INTO Sells VALUES('60001','20004','40003',70);
INSERT INTO Sells VALUES('60002','20004','40002',10);
INSERT INTO Sells VALUES('60003','20000','40001',30);
INSERT INTO Sells VALUES('60004','20001','40000',100);
INSERT INTO Sells VALUES('60005','20003','40003',24);
INSERT INTO Sells VALUES('60006','20003','40000',23);
INSERT INTO Sells VALUES('60007','20002','40002',33);

INSERT INTO HasContract VALUES('70000','30000','20000','James Wood','1/1/10','1/1/20','CONTRACT');
INSERT INTO HasContract VALUES('70001','30000','20003','Burgess Dod','03/01/94','05/31/05','CONTRACT');
INSERT INTO HasContract VALUES('70002','30000','20001','Carleton Cross','02/20/09','07/27/18','CONTRACT');
INSERT INTO HasContract VALUES('70003','30003','20000','Claudia Rayman','05/04/10','01/22/26','CONTRACT');
INSERT INTO HasContract VALUES('70004','30002','20001','Katharine Ciardo','02/26/76','09/02/18','CONTRACT');
INSERT INTO HasContract VALUES('70005','30004','20002','Brit Tour','12/21/04','09/29/23','CONTRACT');
INSERT INTO HasContract VALUES('70006','30000','20002','Rivkah Sherlaw','12/05/77','08/08/84','CONTRACT');
INSERT INTO HasContract VALUES('70007','30003','20002','Esme Blue','11/08/11','03/26/12','CONTRACT');
INSERT INTO HasContract VALUES('70008','30003','20001','Steven Garrettson','03/04/92','04/17/12','CONTRACT');
INSERT INTO HasContract VALUES('70009','30005','20004','Georgine Gaspar','08/30/84','02/13/11','CONTRACT');
INSERT INTO HasContract VALUES('70010','30000','20004','Kalli Urien','02/14/74','02/19/90','CONTRACT');
INSERT INTO HasContract VALUES('70011','30004','20002','Derek Abramovici','04/30/89','02/20/90','CONTRACT');
INSERT INTO HasContract VALUES('70012','30000','20002','Verne Jones','07/17/10','03/16/16','CONTRACT');
INSERT INTO HasContract VALUES('70013','30001','20000','Joseph Bonnette','12/26/85','05/02/18','CONTRACT');
INSERT INTO HasContract VALUES('70014','30005','20000','Nels Janota','03/22/72','03/04/19','CONTRACT');
```

1. select avg(age)
   from Patient natural join Prescription natural join Drugs
   where trade_name = 'Xanax';

   29.5

2. select min(thecount)
   from (select count(phar_co_id) thecount
   from Phar_company natural join HasContract
   group by phar_co_id);

   1

3. select max(thecount)
   from (select count(pat_id) thecount
   from Prescription natural left outer join Patient
   group by pat_id);

   2

4. select trade_name, max(tot_price)
   from (select trade_name, sum(price) tot_price
   from Sells natural join Drugs
   group by trade_name);

   Xanax 173

5. select avg(thecount)
   from (select count(Pharmacy.name) thecount
   from Patient left join Pharmacy on Patient.phar_id = Pharmacy.phar_id
   group by Pharmacy.name);

   1.75

6. Conclusion

This project developed a database design for a drug store chain. It has provided sufficient data and information to provide the drug store chain with a successful manageable database. The users that will be using the database are people with strong medical knowledge. While preparing this software, we had to remember a few things i.e. this software must have a friendly environment, it should have options for future modifications in the database. In this database, all records of daily entry made are stored, so that we can use them in future.

The database design will help with eliminating the vast expenses of using paper based records. Also, the design will help in improving the organization, accuracy, and security of the drug chain database. Upon successful implementation of the project, the drug chain will see an increase in return of investment due to minimal operational costs and successful delivery of services.