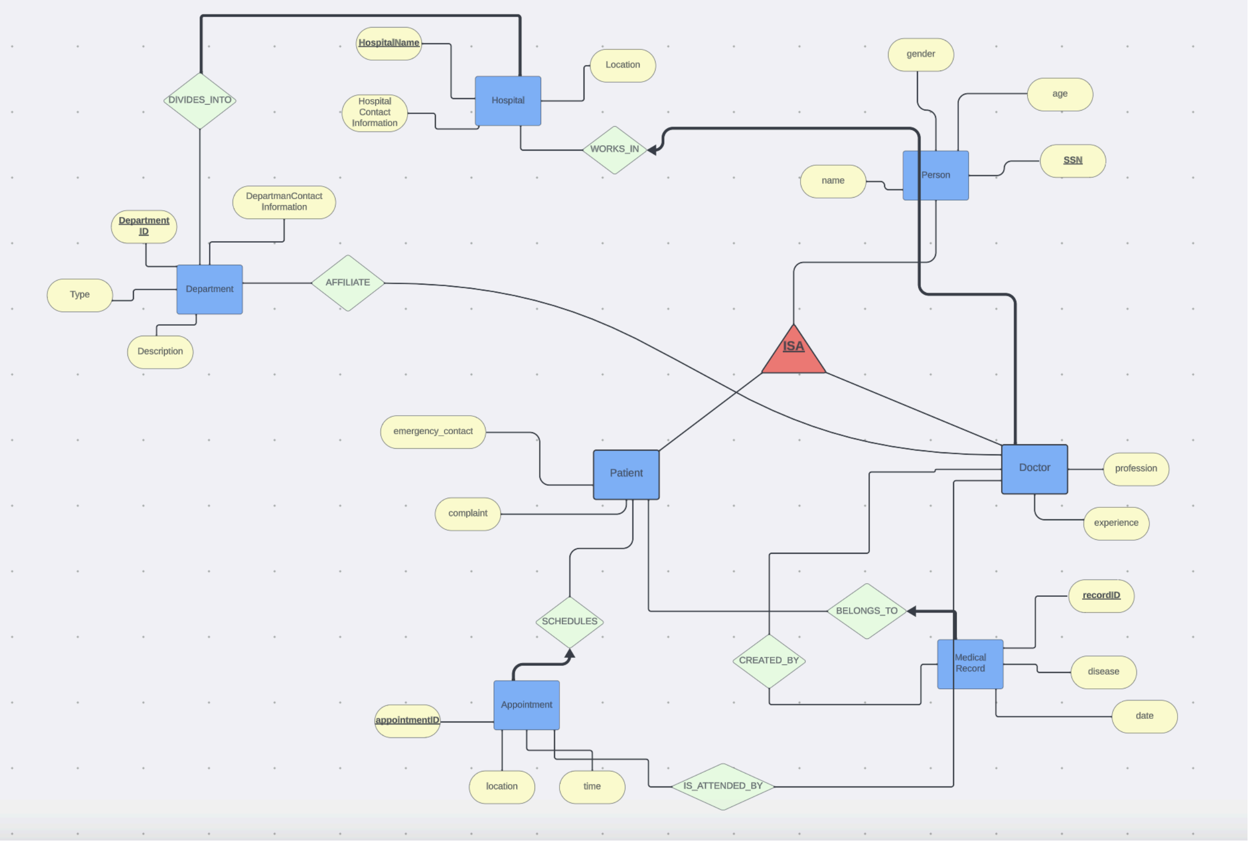
Mert Coşkuner-29120
Demirhan İzer - 29108

HealthDB

In this project, we aim to create a database application that will efficiently organize and manage the information related to hospital work and patient relations. We are going to use 2n+2 entities and employ MYSQL as a database management system. In total, we will have 7 entities which are hospital, department, doctor, patient, person (used as IS-A), appointment, and medical record.

First of all, we have a hospital entity that has 3 attributes, which are location, hospital contact information, and lastly, the unique hospital name and primary key for this entity. Secondly, the department entity has 4 attributes, which are type, department contact information, description, unique department ID, and the primary key for the department with the foreign key hospital name. There will be a Hospital_DIVIDES_INTO (relationship,entity) between hospital and department entities which means that the hospital divides into departments. Moreover, there will be many to many relations, meaning each department can be associated with more than one hospital. Our third entity will be the person entity with 4 attributes, which are name, gender, age, and SSN as a primary key. Also, a person entity has an ISA relationship which means a person must be a doctor or patient. The following entity will be the doctor, which has 2 attributes, which are profession and experience and it has foreign key from Person. Also since this Doctor entity has participation constraint entity and relation is written as whole with WORKS_IN, then it will have PRIMARY KEY as SSN. Furthermore, there will be an AFFILIATE relationship between the doctor and the department. There is a Patient entity that has 3 attributes emergency_contact, complaint, and SSN as primary key for patients. Also, we have an Appointment entity that holds the appointment information of patients so that there is a relation that satisfies schedules relation. This Appointment entity has appointmentid(primary key), time, and location. Additionally, we have an is_attended_by relation that provides a doctor for an appointment. Lastly, we have a MedicalRecord entity that holds the medical records of each patient with attributes of diseases, date, and lastly, the primary key is recordId. In order to store any medical record, there needs to be a belongs_to relation with Patients. Since these medical records need to be created by doctors, they are created_by relation to the Doctor entity.



Underlined and bold→ Primary key
Italic and bold → Foreign key
Thick line → Participation constraint

```
DROP DATABASE IF EXISTS cs306projev1;
CREATE DATABASE cs306projev1;
USE cs306projev1;

create table Person(
  name VARCHAR(100),
  age INTEGER,
  gender VARCHAR(50),
  SSN VARCHAR(50),
  PRIMARY KEY(SSN));

create table Patient(
  Patient_SSN VARCHAR(50),
  emergency_contact VARCHAR(20),
  complaint VARCHAR(150),
  PRIMARY KEY(Patient_SSN),
  FOREIGN KEY (Patient_SSN) REFERENCES Person(SSN) ON DELETE CASCADE);

create table Department(
 departmentID INTEGER,
 contactInformation VARCHAR(50),
 type VARCHAR(50),
 description VARCHAR(50),
 PRIMARY KEY(departmentID));

create table Appointment(
  appointmentID INTEGER,
  appointmentTime DATE,
  location VARCHAR(100),
  PRIMARY KEY(appointmentID));



create table Hospital(
hospital_contactInformation VARCHAR(50),
location VARCHAR(50),
hospitalName VARCHAR(50),
PRIMARY KEY(hospitalName)
);
create table Divides_Into(
  departmentID INTEGER NOT NULL,
  hospitalName VARCHAR(50),
  PRIMARY KEY(hospitalName, departmentID),
  FOREIGN KEY(hospitalName) REFERENCES Hospital(hospitalName) ON DELETE NO
ACTION,
  FOREIGN KEY(departmentID) REFERENCES Department(departmentID) ON DELETE NO
ACTION);


create table Doctor_WORKS_IN (
  profession VARCHAR(70),
  Doctor_SSN VARCHAR(50),
  experience INTEGER,
  hospitalName VARCHAR(50) NOT NULL,
  PRIMARY KEY(Doctor_SSN),
  FOREIGN KEY(Doctor_SSN) REFERENCES Person(SSN) ON DELETE NO ACTION,
  FOREIGN KEY(hospitalName) REFERENCES Hospital(hospitalName) ON DELETE NO
ACTION);
```

```
create table AFFILIATE (
  Doctor_SSN CHAR(11),
  DepartmentID INTEGER,
  PRIMARY KEY(DepartmentID, Doctor_SSN),
  FOREIGN KEY(DepartmentID) REFERENCES Department(DepartmentID) ON DELETE NO
ACTION,
  FOREIGN KEY(Doctor_SSN) REFERENCES Doctor_WORKS_IN(Doctor_SSN) ON DELETE
NO ACTION);


create table Appointment_SCHEDULES (
  Patient_SSN CHAR(11) NOT NULL,
  appointmentID INTEGER,
  appointmentTime DATE,
  location VARCHAR(100),
  FOREIGN KEY(Patient_SSN) REFERENCES Patient(Patient_SSN),
  PRIMARY KEY(appointmentID)
  );



create table IS_ATTENDED_BY (
  appointmentID INTEGER,
  Doctor_SSN VARCHAR(50),
  PRIMARY KEY(appointmentID, Doctor_SSN),
  FOREIGN KEY(Doctor_SSN) REFERENCES Doctor_WORKS_IN(Doctor_SSN) ON DELETE
NO ACTION,
  FOREIGN KEY(appointmentID) REFERENCES Appointment_SCHEDULES(appointmentID)
  );



create table Medical_Record_BELONGS_TO (
  recordID INTEGER NOT NULL,
  disease VARCHAR(100),
  date DATE,
  Patient_SSN CHAR(11) NOT NULL,
  FOREIGN KEY(Patient_SSN) REFERENCES Patient(Patient_SSN) ON DELETE NO
ACTION,
  PRIMARY KEY(recordID));

create table CREATED_BY(
  recordID INTEGER,
  Doctor_SSN VARCHAR(50),
  PRIMARY KEY(recordID, Doctor_SSN),
  FOREIGN KEY(Doctor_SSN) REFERENCES Doctor_WORKS_IN(Doctor_SSN) ON DELETE
NO ACTION,
  FOREIGN KEY(recordID) REFERENCES Medical_Record_BELONGS_TO(recordID) ON
DELETE CASCADE);
```