

CS 207 Database and Information Systems

Aug-Nov 2021

Project Report

All-Inclusive Healthcare Portal

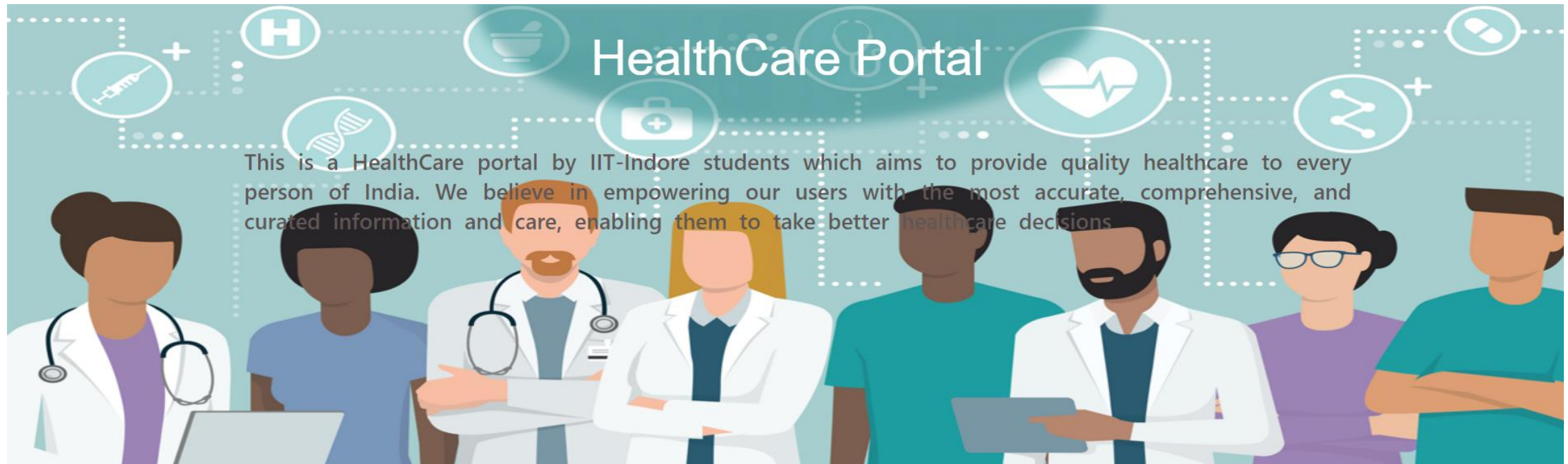
Amit Kumar Makkad - 200001003

Bhavya Contractor - 200001018

Mihir Karandikar - 200001044

Priyansh Jaseja - 200001063

Problem Statement



In this project, we design an All-Inclusive Healthcare Portal. All-Inclusive here signifies that the portal caters to the needs of users, hospitals as well as doctors. Our aim is to create a platform where these three aforementioned entities can interact with each other efficiently resulting in maximum benefit for all parties involved, and consequently contributing to the development of healthcare systems in general.

This project is made as a part of course "Database and Information Systems" in Indian Institute of Technology, Indore.

As in Covid-19 pandemic we have seen that quality HealthCare is one of the basic need of every person so, this project aims to provide quality healthcare to every person.

MySQL Scripts for creating all Tables Present in the Database

1) Hospital Table

```
CREATE TABLE IF NOT EXISTS `Hospital` (  
  `hosp_ID` int,  
  `hospName` varchar(255),  
  `hospEmail` varchar(255),  
  `hospPhone` INT,  
  `hospAddress` varchar(255),  
  `hosp_city` varchar(100),  
  `hosp_state` varchar(100),  
  `hospPassword` varchar(255),  
  `hosp_description` varchar(300),  
  PRIMARY KEY (`hosp_id`));
```

2) Oxygen Details Table

```
CREATE TABLE IF NOT EXISTS `OxygenDetails` (  
  `hosp_ID` INT,  
  `litres_available` INT,  
  `supply_per_hour` INT,  
  `price_per_litre` INT,  
  PRIMARY KEY (`hosp_id`),  
  FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

3) Surgery Details Table

```
CREATE TABLE IF NOT EXISTS `SurgeryDetails` (  
  `hosp_ID` int,  
  `heart` BOOLEAN,  
  `joint` BOOLEAN,  
  `abdominal` BOOLEAN,  
  PRIMARY KEY (`hosp_id`),  
  FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

4) Beds Details Table

```
CREATE TABLE IF NOT EXISTS `BedsDetails` (  
  `hosp_ID` int,  
  `BedsQuantity` int,  
  PRIMARY KEY (`hosp_id`),  
  FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

5) Blood Details Table

```
CREATE TABLE IF NOT EXISTS `BloodDetails` (  
  `hosp_ID` int,  
  `AP_Quantity` int,  
  `AN_Quantity` int,  
  `BP_Quantity` int,  
  `BN_Quantity` int,  
  `OP_Quantity` int,  
  `ON_Quantity` int,  
  `ABP_Quantity` int,  
  `ABN_Quantity` int,
```

```
PRIMARY KEY (`hosp_id`),  
FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

6) Ambulance Details Table

```
CREATE TABLE IF NOT EXISTS `AmbulanceDetails` (  
  `hosp_ID` int,  
  `AmbulanceQuantity` int,  
  PRIMARY KEY (`hosp_id`),  
  FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

7) Doctor Details Table

```
CREATE TABLE IF NOT EXISTS `doctor` (  
  `doc_ID` varchar(100),  
  `docName` varchar(255),  
  `doc_s` varchar(255),  
  `docPhone` INT,  
  `docAddress` varchar(255),  
  `docPassword` varchar(255),  
  `hosp_ID` INT,  
  PRIMARY KEY (`doc_id`),  
  FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

8) User Details Table

```
CREATE TABLE IF NOT EXISTS `user` (  
  `UserId` INT NOT NULL,  
  `UserName` VARCHAR(45) NULL DEFAULT NULL,  
  `UserPassword` VARCHAR(45) NULL DEFAULT NULL,  
  `UserAddress` VARCHAR(45) NULL DEFAULT NULL,  
  `UserPhone` INT NULL DEFAULT NULL,  
  `UserDOB` DATE NULL DEFAULT NULL,  
  PRIMARY KEY (`UserId`));
```

9) Time Slots Table

```
CREATE TABLE IF NOT EXISTS `TimeSlots` (  
  `Time_ID` int auto_increment,  
  `Doc_ID` varchar(100),  
  `Start_Time` time,  
  `End_Time` time,  
  `Appt_Date` date,  
  `Availability` int,  
  PRIMARY KEY (`Time_ID`),  
  FOREIGN KEY (`Doc_id`) REFERENCES Doctor(`doc_ID`)  
);
```

10) Appointment Table

```
CREATE TABLE IF NOT EXISTS `Appointment` (  
  `UserID` int,  
  `Time_ID` int,  
  `doc_ID` varchar(100),  
  `MeetLink` varchar(100),  
  `PreDescription` varchar(100),  
  `PostDescription` varchar(100),  
  `Acceptance_Status` int,  
  PRIMARY KEY (`UserID`,`Time_ID`),  
  FOREIGN KEY (`UserID`) REFERENCES User(`UserID`),  
  FOREIGN KEY (`Time_ID`) REFERENCES TimeSlots(`Time_ID`),  
  FOREIGN KEY (`Doc_id`) REFERENCES Doctor(`doc_ID`)  
);
```

11) Chat Table

```
CREATE TABLE IF NOT EXISTS `chat` (  
  `chat_time` TIME NOT NULL,  
  `sender_id` VARCHAR(45) NOT NULL,  
  `receiver_id` VARCHAR(45) NOT NULL,  
  `msg` VARCHAR(45) NULL DEFAULT NULL,  
  `chat_date` DATE NOT NULL,  
  PRIMARY KEY (`chat_time`, `sender_id`, `receiver_id`, `chat_date`));
```

12) Vaccine Slots Table

```
CREATE TABLE IF NOT EXISTS vaccine_slots (  
    `vaccine_time_id` int auto_increment,  
    `hosp_id` int,  
    `start_time` time,  
    `end_time` time,  
    `appt_date` date,  
    `dose` int,  
    `total_persons` int,  
    `vaccine_type` varchar(100),  
    PRIMARY KEY (`vaccine_time_id`),  
    FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`)  
);
```

13) Vaccine Book Table

```
CREATE TABLE IF NOT EXISTS vaccine_book (  
    `UserID` int,  
    `vaccine_time_id` int,  
    `hosp_id` int,  
    `dose` int,  
    PRIMARY KEY (`dose`, `UserID`),  
    FOREIGN KEY (`hosp_id`) REFERENCES Hospital(`hosp_id`),  
    FOREIGN KEY (`vaccine_time_id`) REFERENCES vaccine_slots(`vaccine_time_id`),  
    FOREIGN KEY (`UserID`) REFERENCES User(`UserID`)  
);
```


14) Doctor Reviews Table

```
CREATE TABLE IF NOT EXISTS `Doctor_Reviews` (  
  `Time_ID` int,  
  `doc_ID` varchar(100),  
  `User_Name` varchar(100),  
  `Review` VARCHAR(100),  
  `Rating` INT,  
  PRIMARY KEY (`Time_ID`),  
  FOREIGN KEY (`Time_ID`) REFERENCES TimeSlots(`Time_ID`),  
  FOREIGN KEY (`doc_ID`) REFERENCES Doctor(`doc_ID`)  
);
```

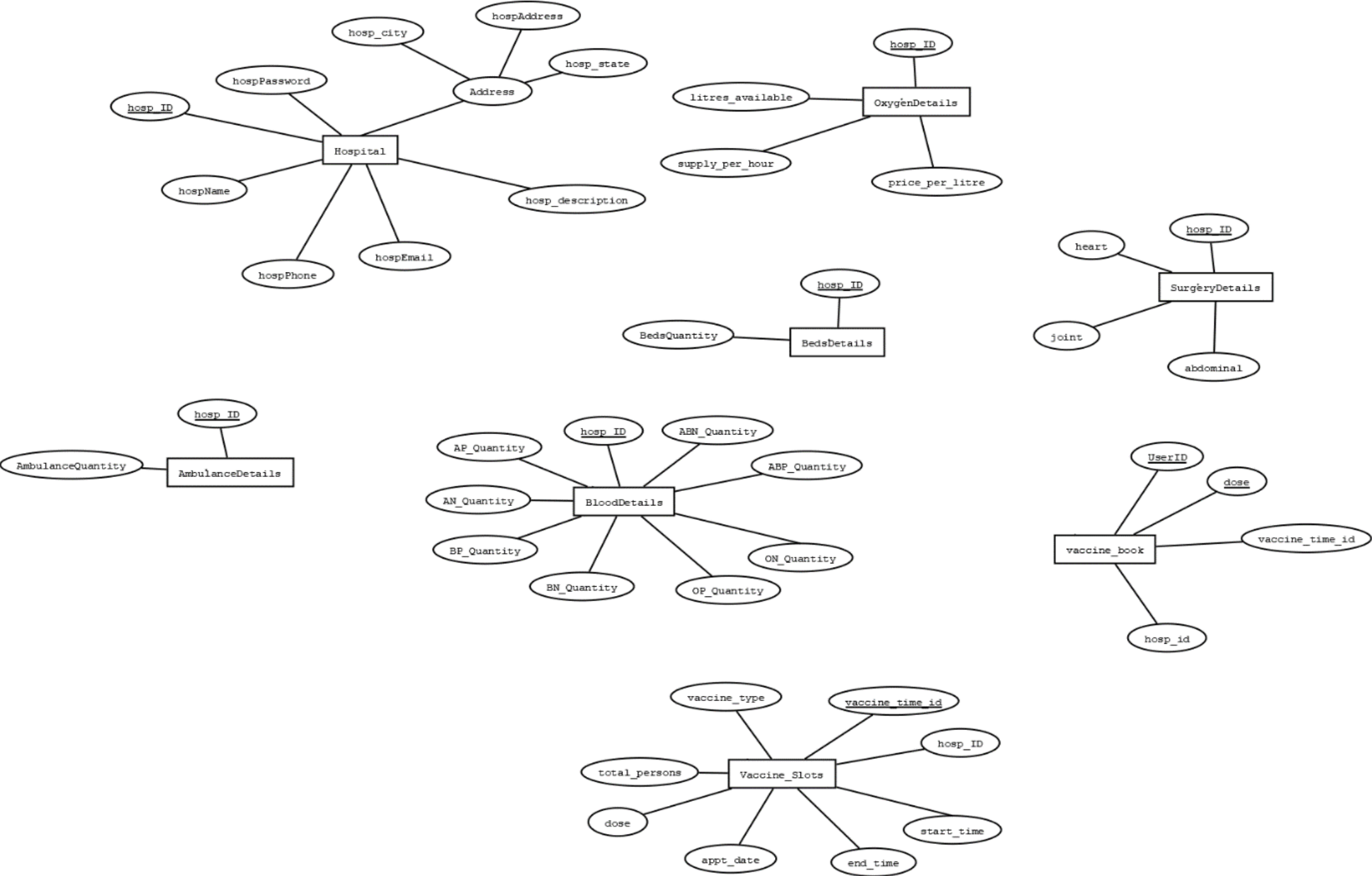
#Acceptance_Status = 0 ... Request has been sent and is pending

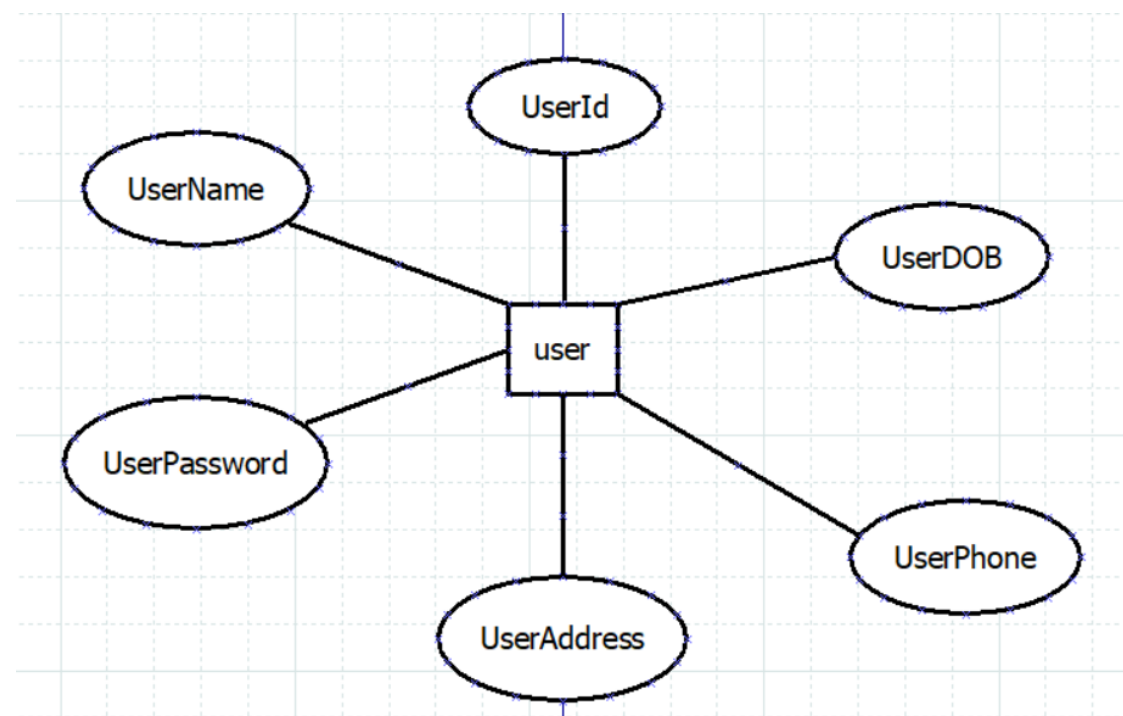
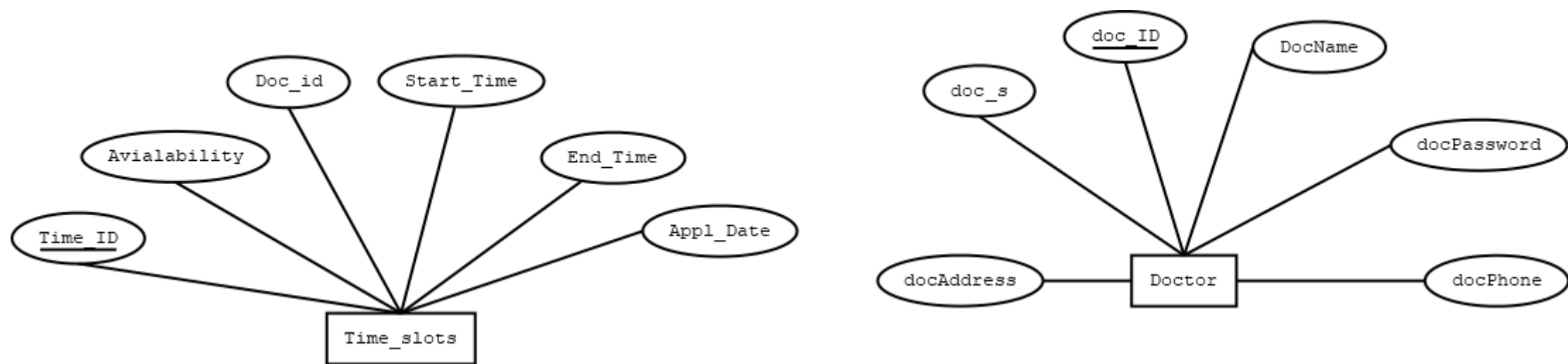
#Acceptance Status = 1 ... Request has been accepted

#Acceptance Status = 2 ... Request has been declined

#Acceptance Status = 3 ... Request has been cancelled

Individual Entities Present in the Database





Entity-Relationship Diagram

