**Queries:**

Please note // = comments

**Login:**

SELECT Type

FROM User

WHERE Username = “$Username” AND Password = “$Password”;

// catch exception if the username and password don't exist in username

// if type is doctor check if the person created a profile

SELECT \*

FROM Doctor

WHERE Username = “$Username”;

// if the result set is null go to profile else go to homepage

// if type is patient check if the person created a profile

SELECT \*

FROM Patient

WHERE Username = “$Username”;

// if the result set is null go to profile else go to homepage

**Register:**

INSERT INTO User (Username, Password, Type)

VALUES (“$Username”, “$Password”, '$Type');

// catch exception if the Username already exists, as being primary key it will through an error which can be caught

**Create/Edit Patient Profile:**

INSERT INTO Patient (Username, AnnualIncome, Name, HomePhone, DateofBirth, Gender, Address, WorkPhone, ContactName, ContactPhone, Weight, Height)

VALUES ("$Username", '$AnnualIncome', "$Name", $HomePhone, $DateOfBirth, '$Gender', "$Address", $Work\_Phone, "$Contact\_Name", $ContactPhone, $Weight, $Height);

ON DUPLICATE KEY

UPDATE AnnualIncome = '$AnnualIncome', Name = "Name", HomePhone = $HomePhone, DateofBirth = $DateofBirth, Gender = '$Gender', Address = "$Address", WorkPhone = $WorkPhone, ContactName = "$ContactName", ContactPhone =$ContactPhone, Weight = $Weight, Height = $Height;

// Inserting an allergy, to insert multiple, send the query with different values multiple times

INSERT INTO Patient2 (UsernameOfPatient, Allergies)

VALUES (“$UsernameOfPatient”, “Allergies");

//For updating allergies can insert new allergies:

INSERT INTO Patient2 (UsernameOfPatient, Allergies)

VALUES (“$UsernameOfPatient”, “Allergies");

//Or can delete existing allergies:

DELETE from Patient2

WHERE UsernameOfPatient = "$UsernameOfPatient" AND Allergies = "$Allergies"

**Create/Edit Doctor Profile**

INSERT INTO Doctor (Username, LicenseNo, FirstName, LastName, DateOfBirth, WorkPhone, Specialty, RoomNumber, HomeAddress)

VALUES ("$Username", "$LicenseNo", "$FirstName","$LastName", $DateOfBirth, $WorkPhone, '$Specialty', $RoomNumber, "$HomeAddress")

ON DUPLICATE KEY

UPDATE FirstName = "$FirstName", LastName = "$LastName", DateOfBirth = $DateOfBirth, WorkPhone = $WorkPhone, Specialty = $Specialty', RoomNumber = $RoomNumber, HomeAddress = "$HomeAddress";

// Insert multiple times the doctor availability times (press the button multiple times in the GUI)

INSERT INTO Doctor\_Availability (Username, Begin, Day, End)

VALUES (“$Username”, $Begin, $Day, $End);

//For availability times can insert new times:

INSERT INTO Doctor\_Availability (Username, Begin, Day, End)

VALUES (“$Username”, $Begin, $Day, $End);

//Can also delete availability times

DELETE FROM Doctor\_Availability

WHERE Username = “$Username” AND Begin = “$Begin” AND End= “$End” AND Day = “$Day”;

**Search Appointments:**

SELECT FirstName, LastName, WorkPhone, Begin, Day, End, x.Rate AS Rating from Doctor

LEFT JOIN

(

SELECT rates.UsernameOfDoctor, avg(Rating) AS Rating From rates

GROUP BY rates.UsernameOfDoctor

) AS x

ON x.UsernameOfDoctor = Doctor.Username

LEFT JOIN Doctor\_Availability

ON Doctor\_Availability.Username = Doctor.Username

WHERE Doctor.Specialty = '$Specialty’;

**Request Appointment:**

INSERT INTO Request\_Appointment (UsernameOfDoctor, UsernameOfPatient, date, scheduledTime)

VALUES ("$UsernameOfDoctor", "$UsernameOfPatient", $date, $scheduledTime);

**Order Medication:**

INSERT INTO Order\_Medication (UsernameOfPatient, MedicineName, Dosage, Duration, Notes)

VALUES ("$UsernameOfPatient", "$MedicineName", $Dosage , $Duration, "$Notes");

**Save Card Info:**

INSERT INTO Payment\_Information (cardNumber, cardType, cvv, cardExpirationDate, cardHolderName, UsernameOfPatient)

VALUES ($cardNumber, "$cardType", $cvv, $cardExpirationDate, "$cardHolderName", "$UsernameOfPatient");

**View visit history:**

// TA said to use dateTime for Date of Visit name of TA was Kaesar on 3/24/2014, as can have multiple visits on the same day, so the theory was // to loop through all possible times (all the seconds) of that day and get all the appointments at that dateTime and put them together in a table

// First generate the visit dates to select one from to see history of that day

SELECT \*

From Visit

WHERE Visit.UsernameOfPatient = “$UsernameOfPatient”;

// once a date is selected the history can be displayed

SELECT \*

FROM Visit

LEFT JOIN Doctor

ON Visit.UsernameOfDoctor = Doctor.Username

LEFT JOIN Visit\_Diagnosis

ON Visit.dateOfVisit = Visit\_Diagnosis.dateOfVisit AND Visit.UsernameOfDoctor = Visit\_Diagnosis.UsernameOfDoctor AND Visit.UsernameOfPatient = Visit\_Diagnosis.UsernameOfPatient

LEFT JOIN Results\_IN

ON Visit.UsernameOfDoctor = Results\_In.UsernameOfDoctor AND Results\_In.UsernameOfPatient = "$UsernameOfPatient" AND Results\_In.dateOfVisit = $dateOfVisit

WHERE Visit.dateOfVisit = $dateOfVisit;

**Rating:**

// first check if patient has visited doctor

SELECT \*

From Visit

LEFT JOIN Doctor

ON Visit.UsernameOfDoctor = Doctor.Username

WHERE UsernameOfPatient = “$UsernameOfPatient”;

// this will be used to generate drop down list from doctor names

// then if a doctor is selected and a rating given the following query is executed.

INSERT INTO rates(UsernameOfDoctor, UsernameOfPatient, Rating)

VALUES (“$Username”, “$Username”, $Rating);

**Appointments Calendar (12):**

// The text said that the patient should not be allowed to make multiple appointments to the same doctor without the doctor’s discretion, //however he/she can request multiple appointments, and whether the visit happens is up to the doctor

SELECT Request\_Appointment .UsernameOfPatient, Request\_Appointment .scheduledTime, Patient.Name

FROM Request\_Appointment

LEFT JOIN Patient

ON Request\_Appointment.UsernameOfPatient = Patient.Username

WHERE Date = $Date

Order BY Request\_Appointment.scheduledTime;

**Appointments Calendar (13):**

//Use loop for all the days of the month to get the count for each day and insert

SELECT Count(scheduledTime)

FROM Request\_Appointment

WHERE date = $date;

**Doctor Views Patient Visit History:**

Select \*

FROM Patient

WHERE Name = "$Name" AND Home\_Phone = $Home\_Phone;

// get details of the person, if doesn’t exist will be caught as an exception in Java

// generate the visit dates to select one from to see history of that day

SELECT \*

FROM Visit

WHERE Visit.UsernameOfPatient = “$UsernameOfPatient” AND Visit.UsernameOfDoctor = “$UsernameOfDoctor”;

// once a date is selected the history can be displayed

SELECT \*

FROM Visit

LEFT JOIN Doctor

ON Visit.UsernameOfDoctor = Doctor.Username

LEFT JOIN Visit\_Diagnosis

ON Visit.dateOfVisit = Visit\_Diagnosis.dateOfVisit AND Visit.UsernameOfDoctor = Visit\_Diagnosis.UsernameOfDoctor AND Visit.UsernameOfPatient = Visit\_Diagnosis.UsernameOfPatient

LEFT JOIN Results\_IN

ON Visit.UsernameOfDoctor = Results\_In.UsernameOfDoctor AND Results\_In.UsernameOfPatient = "$UsernameOfPatient" AND Results\_In.dateOfVisit = $dateOfVisit

WHERE Visit.dateOfVisit = $dateOfVisit;

**Record a New Visit:**

INSERT INTO Visit( UsernameOfDoctor, UsernameOfPatient, DateOfVisit, BillingAmount, Diastolic, Systolic)

VALUES ( “$UsernameOfDoctor”, “$UsernameOfPatient”, $DateOfVisit, $BillingAmount, $Diastolic, $Systolic);

**Record a Surgery:**

// first try and get the the unique and necessary credentials of the person

SELECT Username, Home\_Phone

FROM Patient

WHERE Name = "$Name";

// once the desired person has been selected

INSERT INTO Performs\_Surgery( UsernameOfDoctor, UsernameOfPatient, CPTCode, complications, startTime, endTime, anethesiaStartTime, numberOfAssistants)

VALUES (“$UsernameOfDoctor”, “$UsernameOfPatient”, $CPTCode, “$complications”, $startTime, $sendTime, $anethesiaStartTime, $numberOfAssistants);

**Send Message:**

Doctor – Doctor

INSERT INTO Communicates\_With( UsernameofDoctor, UsernameofDoctor2, content, dateTime, status)

VALUES (“$UsernameofDoctor”, “$UsernameofDoctor2”, “$content”, $dateTime, ‘Unread’);

Patient – doctor

INSERT INTO Send\_Message\_To\_Doctor( UsernameOfDoctor, UsernameOfPatient, content, dateTime, status)

VALUES (“$UsernameOfDoctor”, “$UsernameOfPatient”, “$content”, $dateTime, ‘Unread’);

Doctor – Patient

INSERT INTO Send\_Message\_To\_Patient( UsernameOfDoctor, UsernameOfPatient, content, dateTime, status)

VALUES (“$UsernameOfDoctor”, “$UsernameOfPatient”, “$content”, $dateTime, ‘Unread’);

**View Messages:**

// will have JComboBox to mark as read or unread for which the query will be, assume the dateTime, Username of Doctor, and Username of // User is given from the Result Set

// Please note the doctor will see the messages from patients and messages from other doctors separately this is part of our UI, we can combine // the messages but deemed it unnecessary (join statements have been used above), also in the project it said that we can set the UI however we want it to be

For Patient:

Update Send\_Message\_To\_Patient

SET status = '$status';

WHERE UsernameOfDoctor = “$UsernameOfDoctor” AND UsernameOfPatient = “$UsernameOfPatient” AND dateTime = $dateTime;

For Doctor from Patients:

Update Send\_Message\_To\_Doctor

SET status = '$status';

WHERE UsernameOfDoctor = “$UsernameOfDoctor” AND UsernameOfPatient = “$UsernameOfPatient” AND dateTime = $dateTime;

For Doctor from Doctors:

Update Communicates\_With

SET status = '$status';

WHERE UsernameofDoctor = “$UsernameofDoctor” AND UsernameofDoctor2 = “$UsernameofDoctor2” AND dateTime = $dateTime;

// to get count of unread messages use the count feature in SQL

Patients:

SELECT Count(status) AS Count

FROM Send\_Message\_To\_Patient

WHERE status = ‘Unread’ AND UsernameOfPatient = “$UsernameOfPatient”;

Doctor from Patients:

SELECT Count(status)

FROM Send\_Message\_To\_Doctor

WHERE status = ‘Unread’;

Doctor from Doctors:

SELECT Count(status)

FROM Communicates\_With

WHERE status = ‘Unread’;

// finally we can also get the messages for them each, note the count was computed separately as computing it in a join would require cascaded statements which are unnecessary

For Patient:

SELECT \*

FROM Send\_Message\_To\_Patient

WHERE UsernameOfPatient = “$UsernameOfPatient”;

For Doctors from Patients:

SELECT \*

FROM Send\_Message\_To\_Doctor

WHERE UsernameOfDoctor = “$UsernameOfDoctor”;

For Doctors from Doctors:

SELECT \*

FROM Communicates\_With

WHERE UsernameofDoctor = “$UsernameofDoctor”;