Group: guc353_1
COMP 353, August 10th 2017

Bahamas Sports Physio Center – BSPC Database System

Team

Andy Nguyen	*2733870*
Dinh Duc Bui	*27321651*
Majed Alkhouri	*26089046*
Mehal Patel	*26789234*
Peter Chen	*27391404*

Login Information

URL: https://guc353_1.encs.concordia.ca
Username: guc353_1
Password: mehal420

Table of Contents

Introduction

The Bahamas Sports Physio Center web system allows usage for three types of users. A receptionist, the medical staff members, and patients, for each user they can perform their own tasks.

Database Assumptions

Administration VS. Admin Assumption

1) Only the admin can create, modify or remove workers from the system. Only he or she will have access to this feature, the workers do not.

2) The admin already exists in the system, so he or she can add the workers into the database.

3) There is only one receptionist. The receptionist has a lot of accessibility in the system such as registration, payment or making appointments but he or she will NOT be able to modify, create or delete any information about the workers.

Appointment AND Prescription Assumption

4) The patient who enters the center already has referral from a trainer and is ready to either be registered into the system or have an appointment made.

5) The prescription given to the patient will always be a unique number. No other center can have the same prescription number as another. It is under the assumption that the prescription will be written on a prescription pad and those always have to be unique.

6) Doctors and therapists have coordinated prescription and treatment based on the prescription number given to the patient.

7) Appointments are made by the patient only to receive treatment provided by a therapist. In other words, appointments are not required by the patient to visit a doctor. Patients can only make appointments for themselves.

8) Receptionist can delete or add an appointment, but will not switch any appointment time amongst the patients.

PAYMENTS ASSUMPTION

9) Fees must be paid for the making of appointments, and there shall be no refund for appointment cancellation.

10) Payments should be made on the day of the appointment. Payments made are also assumed to always be in the correct format and amount and in a scenario where the receptionist would give the patient a physical receipt/bill to be payed and inputted through the website.

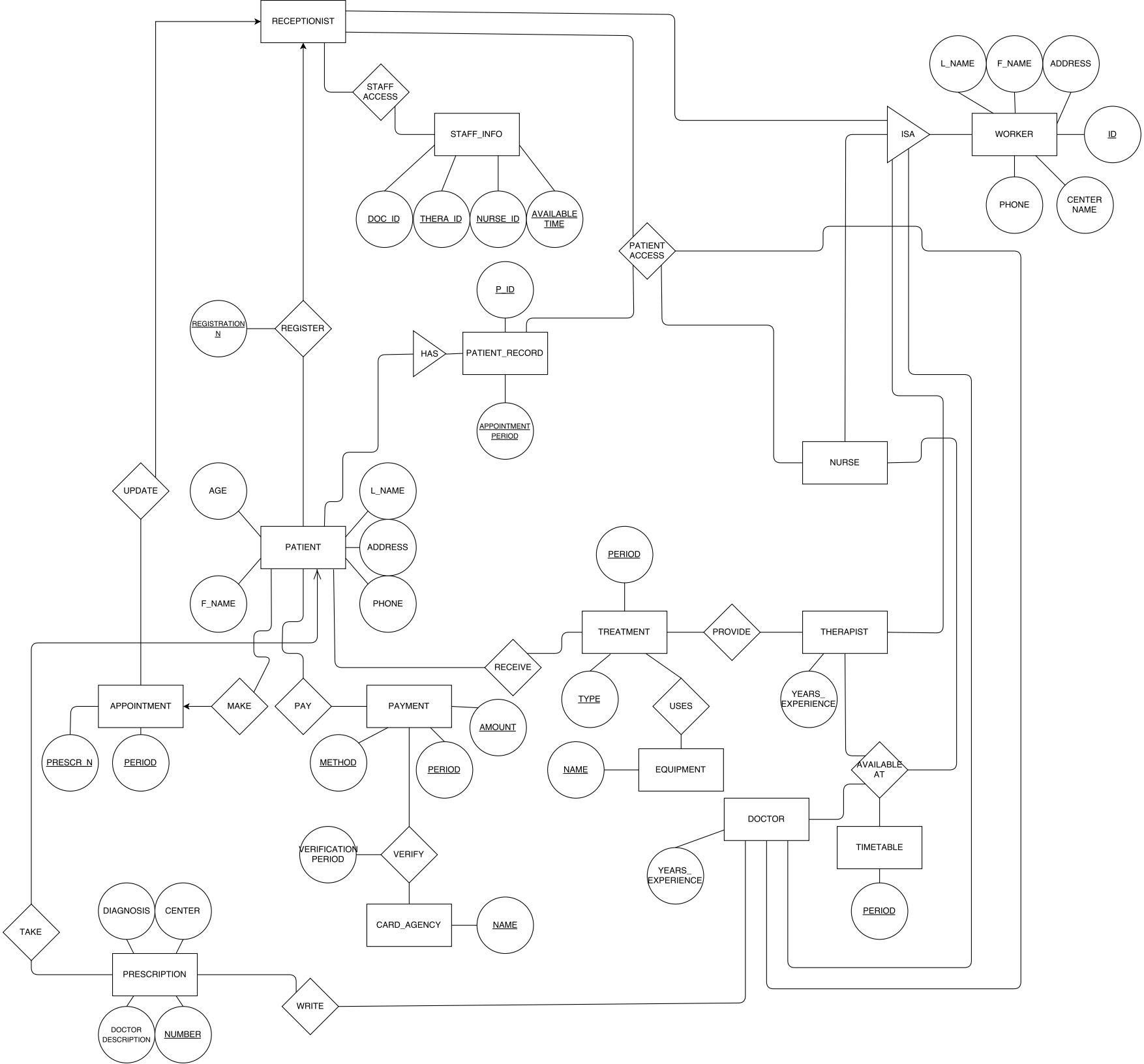
12) Verification of the payments by card processing agency will be done at 21:00:00 everyday.

13) There is only one card processing agency

OTHER ASSUMPTIONS

14) Workers, under the same field of practice, cannot have the same availability time. For instance, two therapists cannot both be free on the same day at 8:00:00.

15) IDs are unique for both workers and patients. In the former case, the first digit of the ID determines the worker's field of practice (i.e. 1=RECEPTIONIST; 2=NURSE; 3=DOCTOR; 4=THERAPIST).


RECEPTIONIST
STAFF ACCESS
STAFF_INFO
DOC_ID
THERA_ID
NURSE_ID
AVAILABLE TIME
L_NAME
F_NAME
ADDRESS
ISA
WORKER
ID
PHONE
CENTER NAME
PATIENT ACCESS
P_ID
REGISTRATION N
REGISTER
HAS
PATIENT_RECORD
APPOINTMENT PERIOD
NURSE
UPDATE
AGE
L_NAME
PATIENT
ADDRESS
F_NAME
PHONE
PERIOD
TREATMENT
PROVIDE
THERAPIST
RECEIVE
TYPE
USES
YEARS_ EXPERIENCE
APPOINTMENT
MAKE
PAY
PAYMENT
AMOUNT
PRESCR_N
PERIOD
METHOD
PERIOD
NAME
EQUIPMENT
AVAILABLE AT
DOCTOR
TIMETABLE
VERIFICATION PERIOD
VERIFY
YEARS_ EXPERIENCE
PERIOD
DIAGNOSIS
CENTER
TAKE
CARD_AGENCY
NAME
PRESCRIPTION
WRITE
DOCTOR DESCRIPTION
NUMBER

Relational Schema Model

/******************RELATIONS CONVERTED FROM ENTITY SETS**********************/

APPOINTMENT(PRESCR N, PERIOD)
- Primary Key= PERIOD
- Foreign Key:
    - PRESCR_N, which references PRESCRIPTION(NUMBER)
    - PERIOD, which references PROVIDE(PERIOD)

CARD_AGENCY(NAME)
- Primary Key= NAME

DOCTOR(ID, YEARS_EXPERIENCE)
- Primary Key= ID
- Foreign Key= ID, which references worker(ID)

EQUIPMENT(NAME)
- Primary Key= NAME

NURSE(ID)
- Primary Key= ID
- Foreign Key= ID, which references worker(ID)

PATIENT(P_ID, F_NAME, L_NAME, AGE, ADDRESS, PHONE)
- Primary Key= P_ID
- Foreign Key= P_ID, which references patient_record(P_ID)

PAYMENT(METHOD, PERIOD, AMOUNT)
- Primary Key= AMOUNT
- Foreign Key=PERIOD, which references APPOINTMENT(PERIOD)

PRESCRIPTION(NUMBER, DIAGNOSIS, CENTER, DOCTOR_ DESCRIPTION)
- Primary Key= NUMBER
- Foreign Key=NUMBER which references WRITES(NUMBER)

RECEPTIONIST(ID)
- Primary Key= ID
- Foreign Key= ID, which references worker(ID)

STAFF_INFO(DOC_ID, THERA_ID, NURSE_ID, AVAILABLE TIME)
- Primary Key= AVAILABLE_TIME
- Foreign Key:
    - DOC_ID, which references DOCTOR(ID)
    - THERA_ID, which references THERAPIST(ID)
    - NURSE_ID, which references NURSE(ID)
    - AVAILABLE_TIME, which references TIMETABLE(PERIOD)

THERAPIST(ID, YEARS_EXPERIENCE)
- Primary Key= ID
- Foreign Key= ID, which references worker(ID)

TIMETABLE(PERIOD)

- Primary Key= PERIOD

TREATMENT(PERIOD, TYPE)
- Primary Key=PERIOD,
- Foreign Key=PERIOD, which references TIMETABLE(PERIOD)

patient_record(P_ID, APPOINTMENT PERIOD, THERA_ID)
- Primary Key= P_ID
- Foreign Key:
    - APPOINTMENT_PERIOD, THERA_ID, which reference PROVIDE(PERIOD, THERA_ID)

worker(ID, F_NAME, L_NAME, ADDRESS, CENTER NAME, PHONE)
- Primary Key= ID

/**********RELATIONS CONVERTED FROM RELATIONSHIP SETS**********/

AVAILABLE_AT(THERA_ID, DOC_ID, NURSE_ID, PERIOD)
- Primary Key= PERIOD
- Foreign Key:
    - PERIOD, which references TIMETABLE(PERIOD)
    - THERA_ID, which references THERAPIST(ID)
    - DOC_ID, which references DOCTOR(ID)
    - NURSE_ID, which references NURSE(ID)

MAKE(PRESCR N, PERIOD, P_ID)
- Primary Key= PRESCR_N
- Foreign Key:
    - PERIOD, PRESCR_N which reference APPOINTMENT(PERIOD,PRESCR_N)
    - P_ID, which references PATIENT(P_ID)

PATIENT_ACCESS(NURSE_ID, DOC_ID, R_ID, P_ID)
- Primary Key= P_ID
- Foreign Key:
    - P_ID, which references patient_record(P_ID)
    - R_ID, which references RECEPTIONIST(ID)
    - DOC_ID, which references DOCTOR(ID)
    - NURSE_ID, which references NURSE(ID)

PAY(P_ID, AMOUNT, PERIOD)
- Primary Key=PERIOD,
- Foreign Key:
    - P_ID, which references PATIENT(P_ID)
    - AMOUNT, PERIOD which references PAYMENT(AMOUNT, PERIOD)

PROVIDE(PERIOD, TYPE, THERA_ID)
- Primary Key= THERA_ID,
- Foreign Key:
    - PERIOD, TYPE which references TREATMENT(PERIOD,TYPE)

- THERA_ID, which references THERAPIST(ID)

RECEIVE(PERIOD, TYPE, P_ID)
- Primary Key= P_ID
- Foreign Key:
  - PERIOD, TYPE which references PROVIDE(PERIOD,TYPE)
  - P_ID, which references PATIENT(P_ID)

REGISTER(REGISTRATION_N, R_ID, P_ID)
- Primary Key= REGISTRATION_N
- Foreign Key:
  - P_ID, which references PATIENT(P_ID)
  - R_ID, which references RECEPTIONIST(ID)

STAFF_ACCESS(AVAILABLE_TIME, R_ID)
- Primary Key= R_ID
- Foreign Key:
  - R_ID, which references RECEPTIONIST(ID)
  - AVAILABLE_TIME, which references TIMETABLE(PERIOD)

TAKE(P_ID, NUMBER)
- Primary Key= NUMBER
- Foreign Key:
  - P_ID, which references PATIENT(P_ID)
  - NUMBER, which references PRESCRIPTION(NUMBER)

UPDATES(R_ID, APPOINTMENT_PERIOD, PRESCR_N)
- Primary Key= R_ID
- Foreign Key:
  - R_ID which references from RECEPTIONIST(ID)
  - APPOINTMENT_PERIOD, PRESCR_N which reference APPOINTMENT(PERIOD, PRESCR_N)

USES(TREAT_PERIOD, TREAT_TYPE, EQUIP_NAME)
- Primary Key= EQUIP_NAME
- Foreign Key:
  - TREAT_PERIOD, TREAT_TYPE which reference from TREATMENT(PERIOD,TYPE)
  - EQUIP_NAME, which references EQUIPMENT(NAME)

VERIFY(VERIFICATION_PERIOD, CARD_AGENCY_NAME, PAY_AMOUNT, PAY_PERIOD)
- Primary Key= PAY_PERIOD
- Foreign Key:
  - CARD_AGENCY_NAME, which references CARD_AGENCY(NAME)
  - PAY_AMOUNT, PAY_PERIOD, which reference PAYMENT(AMOUNT, PERIOD)

WRITES(DOC_ID, NUMBER)
- Primary Key= NUMBER
- Foreign Key:

- DOC_ID, which references DOCTOR(ID)

Functional Dependencies

For the relation to be in 3NF, one of the following three properties must be filled: must be trivial; LHS is a superkey; or if X-> A then A is part of the key.

STAFF_INFO has four elements which are all keys, therefore the FDs are trivial

STAFF_INFO(Doc_ID, Thera_ID, Nurse_ID, Available Time)
FDs= {(Doc_ID, Thera_ID, Nurse_ID, Available Time -> Doc_ID), (Doc_ID, Thera_ID, Nurse_ID, Available Time -> Thera_ID), (Doc_ID, Thera_ID, Nurse_ID, Available Time -> Nurse_ID), (Doc_ID, Thera_ID, Nurse_ID, Available Time -> Avalible_Time)}

Patient_record has elements which again are all keys therefore the FD is trivial

patient_record(P_ID, Appointment_Period)
FDs= {(P_ID, Appointment Period-> P_ID), (P_ID, Appointment_Period -> Appointment Period)}

PATIENT relation has one key P_ID, which is a superkey as it can obtain all the other elements.

PATIENT (P_ID, F_Name, L_Name, Age, Address, Phone)
FDs= {(P_ID -> F_Name), (P_ID -> L_Name), (P_ID -> Age), (P_ID -> Address), (P_ID > Phone)}

APPOINTMENT has two keys Presc_N and Period, and they are the only two elements therefore the FD is trivial.

APPOINTMENT (Presc_N, Period)
FDs= {(Presc_N, Period -> Presc_N), (Presc_N, Period -> Period)}

PAYMENT has 3 elements which are all keys therefore it is trivial.

PAYMENT (Method, Period, Amount)
FDs= {(Method, Period, Amount -> Method), (Method, Period, Amount -> Period), (Method, Period, Amount -> Amount)}

CARD_AGENCY has two elements which are both boys therefore it is trivial.

CARD_AGENCY (Period, Amount)
FDs= {(Period, Amount -> Period), (Period, Amount -> Amount)}

PRESCRIPTION has one key, Number, which is a super key and can retrieve all the other elements.

PRESCRIPTION(Number, Diagnosis, Center, Doctor_Description)
FDs= {(Number -> Diagnosis), (Number-> Center), (Number -> Doctor_Description)}

TREATMENT has only two keys Period and Type which are both candidate keys. Therefore, the FD is trivial

TREATMENT (Period, Type)
FDs= {(Period, Type –> Period), (Period, Type -> Type)}

EQUIPMENT has only one element and it is the key, therefore the FD is trivial

EQUIPMENT (Name)
FDs= {(Name –> Name)}

TIMETABLE has only one element and it is the key, therefore it is trivial

TIMETABLE (Period)
FDs= {(Period -> Period)}

Worker has many elements but has one key ID which is a super key. The ID can retrieve all other elements of worker. Worker is also the parent for Receptionist, Nurse, Doctor and Therapist which all follow the same relations as Worker.

worker(ID, F_Name, L_Name, Address, Center_Name, Phone)
FDs= {(ID -> F_Name, ID -> L_Name), (ID -> Address), (ID -> Center_Name), (ID -> Phone)}

*NOTE: The FDs relative to the relational schema derived from the relationship sets are all trivial. Hence, they all satisfy the 3NF requirement.

Reports + Additional Reports

REPORT 1. How many patients has each physiotherapists seen in a specified period of time?

```
mysql> SELECT COUNT(P_ID), THERA_ID
    -> FROM RECEIVE e, PROVIDE p
    -> WHERE e.PERIOD = p.PERIOD AND p.THERA_ID = '40001';
+-------------+----------+
| COUNT(P_ID) | THERA_ID |
+-------------+----------+
|           1 |    40001 |
+-------------+----------+
1 row in set (0.00 sec)

mysql>
```

/**/

REPORT 2. Which piece of equipment has never been used?

```
mysql> SELECT NAME
    -> FROM USES, EQUIPMENT
    -> WHERE NAME NOT IN (SELECT EQUIP_NAME FROM EQUIPMENT
    -> JOIN USES ON NAME = EQUIP_NAME) GROUP BY NAME;
+--------------------+
| NAME               |
+--------------------+
| EXERCISE BIKE      |
| ISOKINETIC DEVICE  |
| NORDIC TRACK       |
| STATIONARY BICYCLE |
| TREADMILL          |
| WEIGHTS            |
+--------------------+
6 rows in set (0.00 sec)

mysql>
```

/***/

REPORT 3. List all the information available for physio patients who have been at the center.

```
mysql> SELECT P_ID, F_NAME, L_NAME, ADDRESS, PHONE, AGE
    -> FROM PATIENT;
+------+-------------+-------------+-------------------+---------------+------+
| P_ID | F_NAME      | L_NAME      | ADDRESS           | PHONE         | AGE  |
+------+-------------+-------------+-------------------+---------------+------+
|    2 | RAGUEL      | PHILYAW     | 2000 STREET TWO   | (514)111-2222 |   19 |
|    3 | RHEDWIG     | DINEEN      | 3000 STREET THREE | (514)111-3333 |   21 |
|    4 | KRISTIAN    | FULLAM      | 4000 STREET FOUR  | (514)111-4444 |   48 |
|    5 | ORALEE      | LUNDREN     | 5000 STREET FIVE  | (514)111-5555 |   55 |
|    6 | JANETT      | BILLINGSLEA | 6000 STREET SIX   | (514)111-6666 |   33 |
|    7 | KAZMIERCZAK | FLORIA      | 7000 STREET SEVEN | (514)111-7777 |   24 |
|    8 | WEI         | BOOMER      | 8000 STREET EIGHT | (514)111-8888 |   34 |
|   10 | DR          | DRE         | 12 Gangster       | (514)333-3524 |   30 |
|   11 | Dumb        | Dumb        | 123               | 514           |   23 |
|   12 | Andy        | Ng          | 123               | 123           |   55 |
|   13 | Bob         | Patel       | dd                | 231           |   63 |
|   14 | Testy       | Test        | 123 Imagine       | 5147894561    |   96 |
+------+-------------+-------------+-------------------+---------------+------+
12 rows in set (0.00 sec)
```

/***/

REPORT 4. List all the information available for therapists who have been at the center.

```
mysql> SELECT F_NAME, L_NAME, ADDRESS, PHONE, CENTER, ID
    -> FROM worker
    -> WHERE ID > 40000 AND CENTER <> 'BSPC';
+---------+-----------+-----------------+---------------+---------------+-------+
| F_NAME  | L_NAME    | ADDRESS         | PHONE         | CENTER        | ID    |
+---------+-----------+-----------------+---------------+---------------+-------+
| INGA    | ALVARES   | 404 STREET 4PA4 | (514)403-5254 | ABBC          | 40004 |
| DNIE    | CHATTERTON| 405 STREET 4PA5 | (514)405-5255 | ABBC          | 40005 |
| SHERIE  | WALCOTT   | 406 STREET 4PA6 | (514)406-5256 | ABBC          | 40006 |
| PAMELIA | HIGH      | 407 STREET 4PA7 | (514)407-5257 | BBC           | 40007 |
| ERZA    | SWAYNE    | 408 STREET 4PA8 | (514)408-5258 | BBC           | 40008 |
| WEI     | KOOPMAN   | 409 STREET 4PA9 | (514)409-5259 | POKEMON CENTER| 40009 |
| ROGER   | RASKIN    | 410 STREET 4PA10| (514)410-5260 | LHSC          | 40010 |
+---------+-----------+-----------------+---------------+---------------+-------+
7 rows in set (0.00 sec)
```

/***/

REPORT 5. List all the information available for therapists who work at the center.

```
mysql> SELECT F_NAME, L_NAME, ADDRESS, PHONE, CENTER, ID
    -> FROM worker
    -> WHERE ID > 40000 and CENTER = 'BSPC';
+-----------+-----------+-----------------+---------------+--------+-------+
| F_NAME    | L_NAME    | ADDRESS         | PHONE         | CENTER | ID    |
+-----------+-----------+-----------------+---------------+--------+-------+
| ROSA      | BICKHAM   | 401 STREET 4PA1 | (514)401-5251 | BSPC   | 40001 |
| GAYLORD   | CURTSINGER| 402 STREET 4PA2 | (514)402-5252 | BSPC   | 40002 |
| FLORENCIO | VEILLEUX  | 403 STREET 4PA3 | (514)403-5253 | BSPC   | 40003 |
+-----------+-----------+-----------------+---------------+--------+-------+
3 rows in set (0.00 sec)

mysql>
```

/**/

REPORT 6. List detail of reservations for a specific patient.

```
mysql> SELECT P_ID, TYPE, PERIOD FROM RECEIVE
    -> WHERE P_ID = '2';
+------+--------------------+---------------------+
| P_ID | TYPE               | PERIOD              |
+------+--------------------+---------------------+
|    2 | ULTRASOUND MASSAGE | 2017-09-02 14:30:00 |
+------+--------------------+---------------------+
1 row in set (0.00 sec)
```

/**/

REPORT 7. List availability for physiotherapist/doctor during a specified period of time.

```
mysql> SELECT DOC_ID, THERA_ID, PERIOD
    -> FROM AVAILABLE_AT
    -> WHERE PERIOD BETWEEN "2017-09-01" AND "2017-09-03";
+--------+----------+---------------------+
| DOC_ID | THERA_ID | PERIOD              |
+--------+----------+---------------------+
|   NULL |    40001 | 2017-09-01 01:12:12 |
|   NULL |    40001 | 2017-09-01 01:12:15 |
|  30001 |    40001 | 2017-09-01 14:00:00 |
|  30001 |    40001 | 2017-09-01 14:30:00 |
|  30001 |    40001 | 2017-09-01 15:00:00 |
|  30002 |    40002 | 2017-09-02 13:00:00 |
|  30002 |    40002 | 2017-09-02 13:30:00 |
|  30002 |    40002 | 2017-09-02 14:00:00 |
|  30002 |    40002 | 2017-09-02 14:30:00 |
+--------+----------+---------------------+
9 rows in set (0.00 sec)
```

/**/

EXTRA REPORTS

Years of experience from doctor:

```
mysql> SELECT ID, YEARS_EXPERIENCE
    -> FROM DOCTOR;
+-------+------------------+
| ID    | YEARS_EXPERIENCE |
+-------+------------------+
| 30001 |               12 |
| 30002 |                3 |
| 30003 |                8 |
| 30004 |                1 |
| 30005 |                2 |
| 30006 |                3 |
| 30007 |                4 |
| 30008 |                5 |
| 30009 |                9 |
| 30010 |               10 |
+-------+------------------+
10 rows in set (0.00 sec)
```

Years of experience from therapist report:

```
mysql> SELECT ID, YEARS_EXPERIENCE
    -> FROM THERAPIST;
+-------+------------------+
| ID    | YEARS_EXPERIENCE |
+-------+------------------+
| 40001 |                5 |
| 40002 |                6 |
| 40003 |                7 |
| 40004 |                8 |
| 40005 |                9 |
| 40006 |               10 |
| 40007 |               11 |
| 40008 |                8 |
| 40009 |                6 |
| 40010 |                1 |
+-------+------------------+
10 rows in set (0.00 sec)
```

List of all staff information accessed by the receptionist:

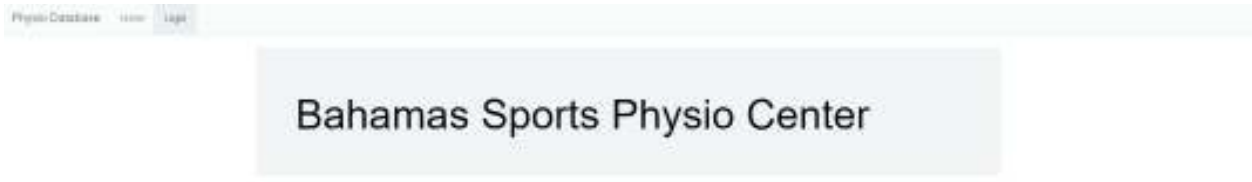
```
mysql> SELECT DOC_ID, THERA_ID, NURSE_ID, AVAILABLE_TIME
    -> FROM STAFF_INFO;
+--------+----------+----------+---------------------+
| DOC_ID | THERA_ID | NURSE_ID | AVAILABLE_TIME      |
+--------+----------+----------+---------------------+
|  30001 |    40001 |    20001 | 2017-09-01 14:00:00 |
|  30001 |    40001 |    20001 | 2017-09-01 14:30:00 |
|  30001 |    40001 |    20001 | 2017-09-01 15:00:00 |
|  30002 |    40002 |    20002 | 2017-09-02 13:00:00 |
|  30002 |    40002 |    20002 | 2017-09-02 13:30:00 |
|  30002 |    40002 |    20002 | 2017-09-02 14:00:00 |
|  30002 |    40002 |    20002 | 2017-09-02 14:30:00 |
|  30003 |    40003 |    20003 | 2017-09-03 15:00:00 |
|  30003 |    40003 |    20003 | 2017-09-03 15:15:00 |
|  30003 |    40003 |    20003 | 2017-09-03 15:30:00 |
|  30003 |    40003 |    20003 | 2017-09-03 15:45:00 |
|  30004 |    40004 |    20004 | 2017-09-04 16:00:00 |
|  30004 |    40004 |    20004 | 2017-09-04 16:15:00 |
|  30005 |    40005 |    20005 | 2017-09-05 13:15:00 |
|  30005 |    40005 |    20005 | 2017-09-05 14:15:00 |
|  30006 |    40006 |    20006 | 2017-09-06 14:15:00 |
|  30006 |    40006 |    20006 | 2017-09-06 14:35:00 |
|  30006 |    40006 |    20006 | 2017-09-06 14:50:00 |
|  30007 |    40007 |    20007 | 2017-09-07 13:00:00 |
|  30007 |    40007 |    20007 | 2017-09-07 13:20:00 |
|  30007 |    40007 |    20007 | 2017-09-07 13:50:00 |
|  30008 |    40008 |    20008 | 2017-09-08 09:10:00 |
|  30008 |    40008 |    20008 | 2017-09-08 10:10:00 |
|  30009 |    40009 |    20009 | 2017-09-09 09:20:00 |
|  30009 |    40009 |    20009 | 2017-09-09 09:50:00 |
|  30010 |    40010 |    20010 | 2017-09-10 09:30:00 |
|  30010 |    40010 |    20010 | 2017-09-10 10:30:00 |
+--------+----------+----------+---------------------+
```

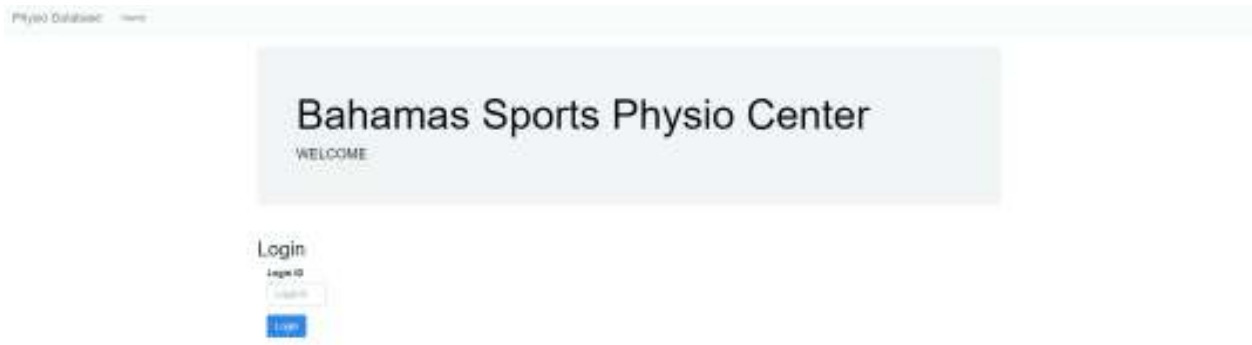
List of patient payment methods:

```
mysql> SELECT METHOD FROM PAYMENT GROUP BY METHOD;
+--------+
| METHOD |
+--------+
| CARD   |
| CASH   |
| CHEQUE |
+--------+
3 rows in set (0.00 sec)
```

Sample Session

Main Page





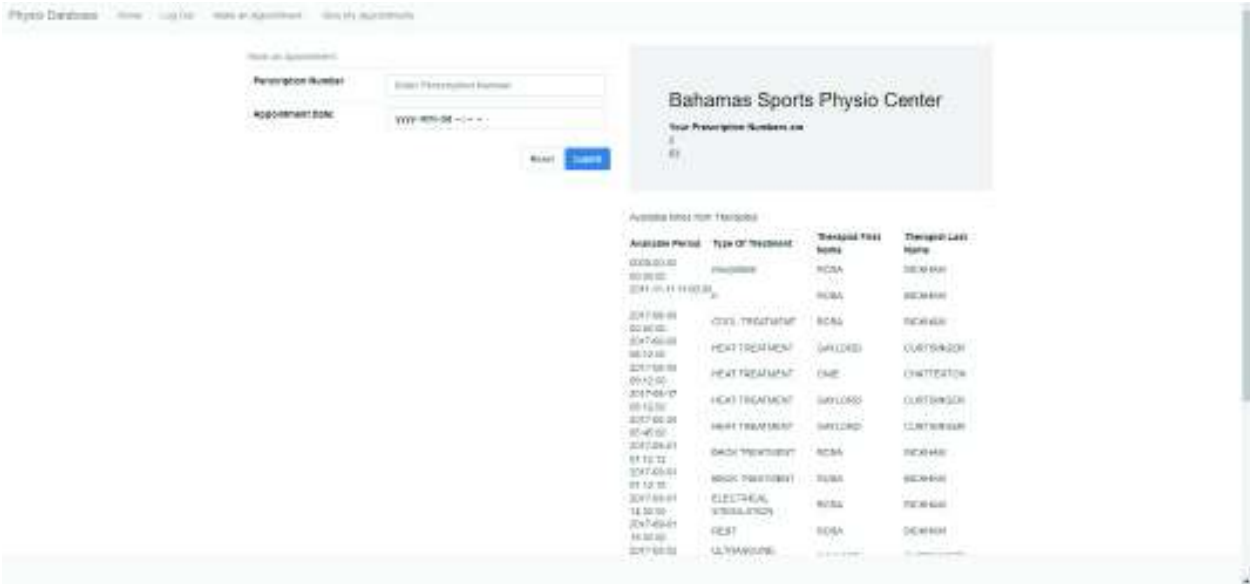
-The user will arrive at the home page(top image) and then he/she will move on to the login page where they will login in with their ID.

Patient

Home Page:

Once successfully logged in, the patient will be greeted to this page where there are two options take.

Make Appointment:



-Here the user can enter his/her prescription number obtained from the Doctor and can enter an appointment date.

View Appointment:

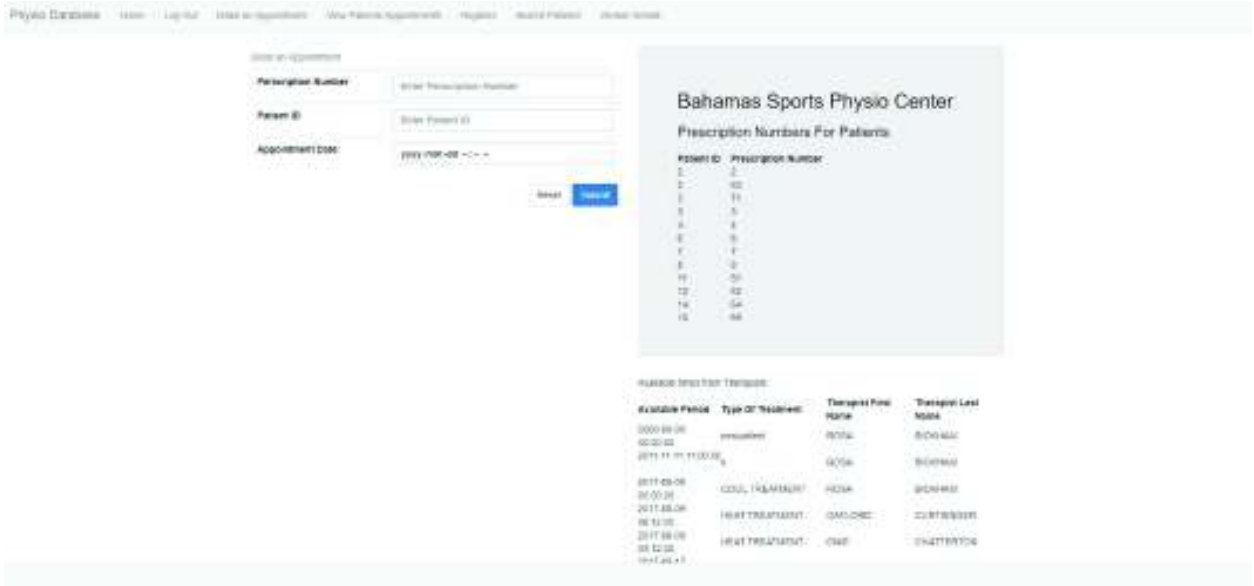
-The patient can view his/her appointments that they made

Receptionist

Receptionist HomePage:

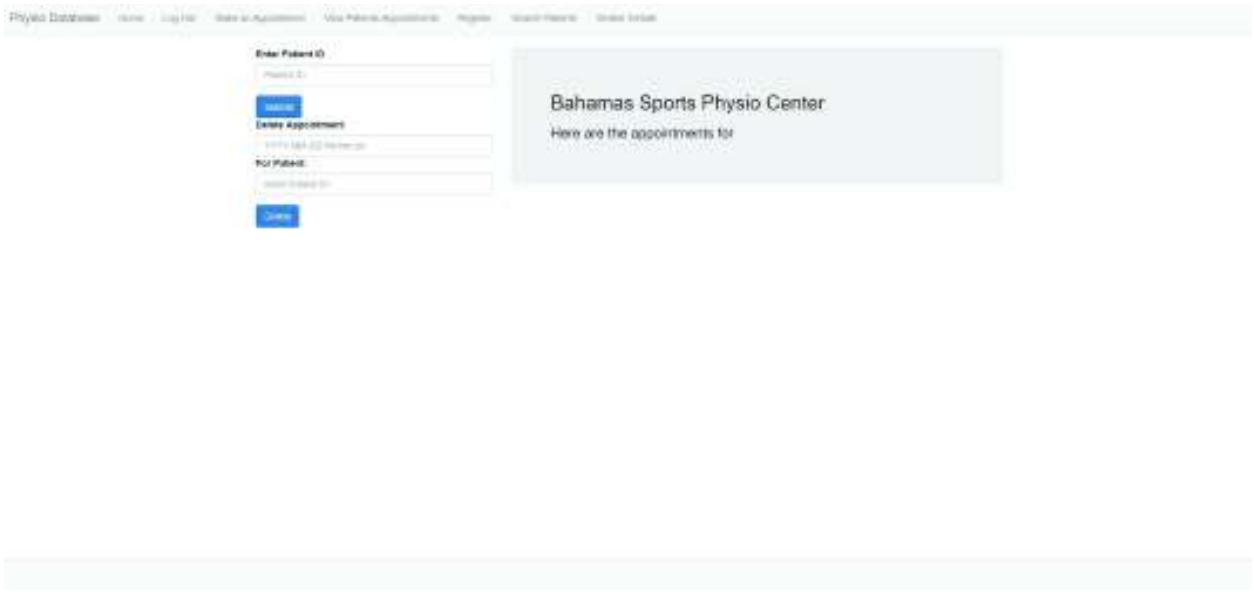
-The receptionist will be greeted by this page once successfully logged in, and from here he/she has many other option in the header.

Make Appointment:



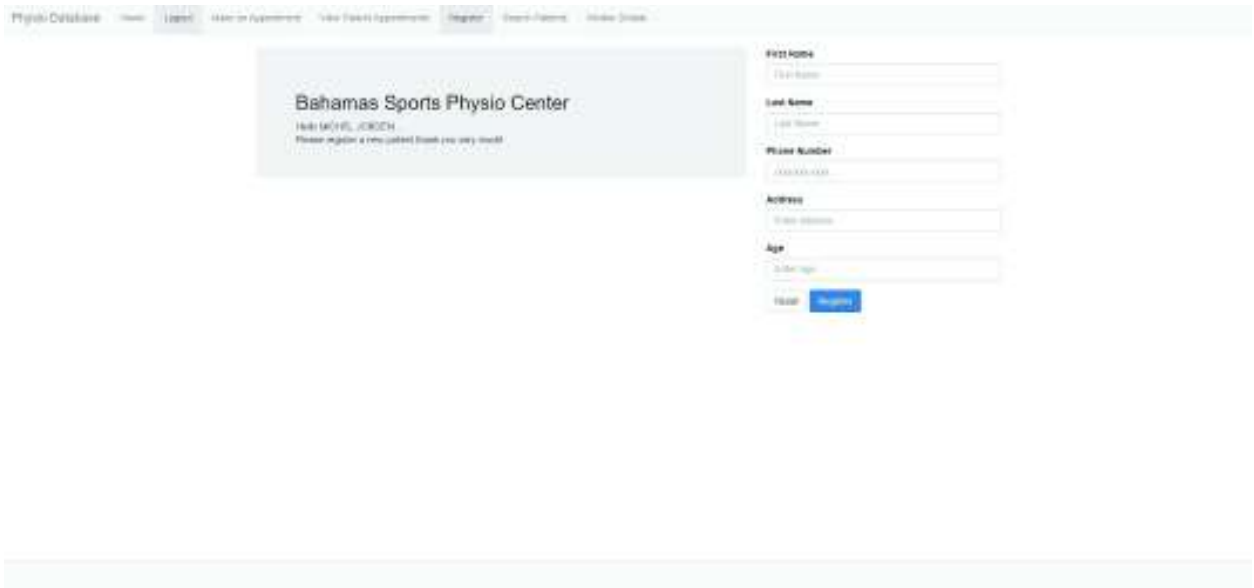
-The receptionist can make an appointment for a patient, by entering prescription number, the patient ID and choosing from the available time slots.

View/Modify Appointments:



-The receptionist at this page can look up a patient's appointments OR can delete an appointment entry by entering the proper fields.

Register Patient:

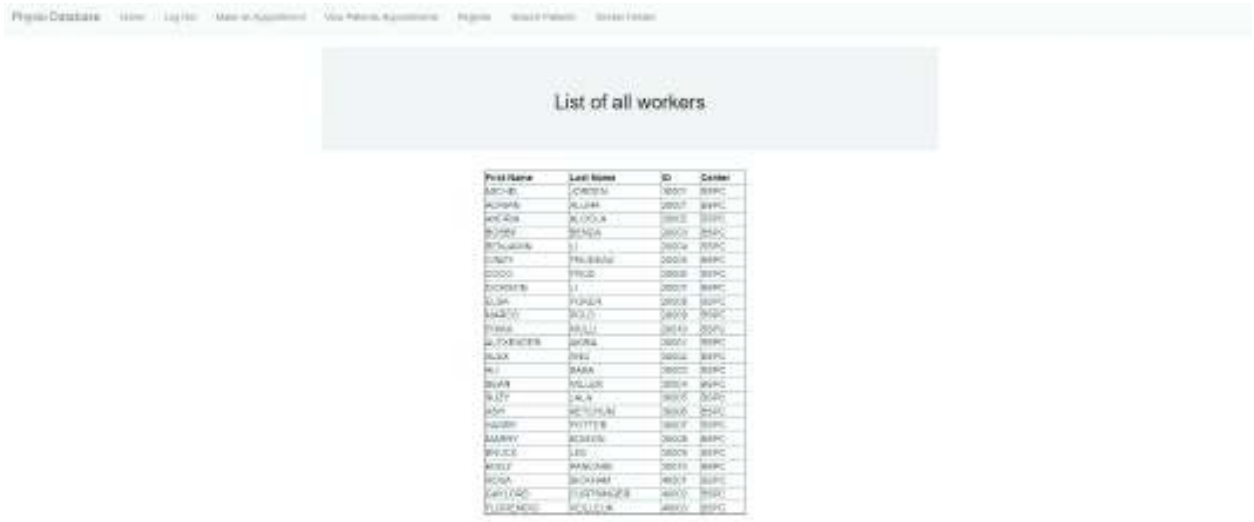


-The receptionist can register a new patient on this page once entering all the necessary fields and a unique ID will be given to the patient.

Search Patients:

-Here the receptionist can search up the patient and patient details such as name, ID, and records will be shown.

Workers List:



-The receptionist has a list of all workers in the the center

Doctor

Doctor Home Page:

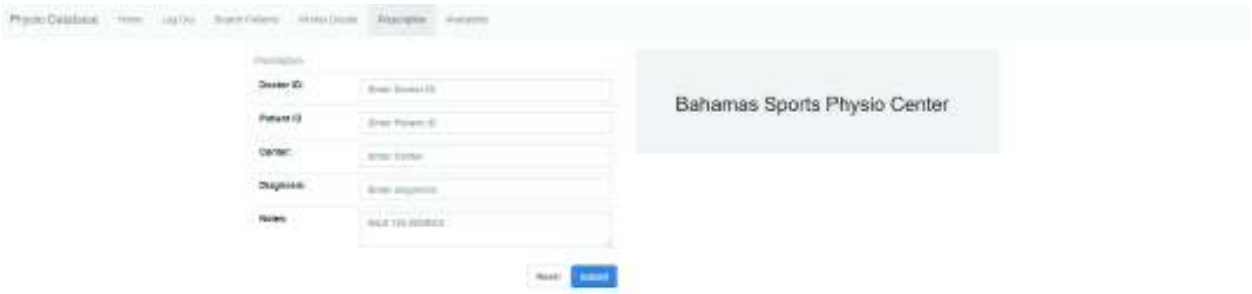
-Doctor will be greeted with this home page when logged in

Doctor Search:

-The doctor can look up patient records from this page with the correct first name entered

The doctor also has a list of all the workers in the center and look exactly like the one from receptionist.

Doctor Prescription:



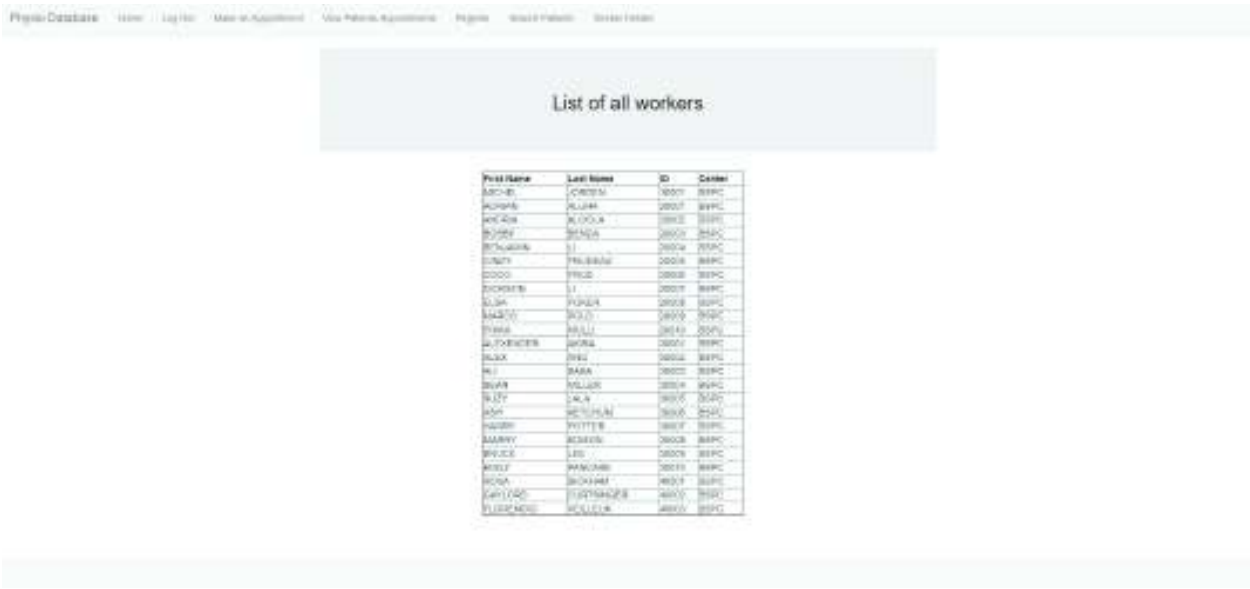
-The doctor enters all the information into the field; such as doctor ID, patient ID, the center, diagnosis and a short description and it will return a unique prescription number that the patient will use to make an appointment with a therapist.

-Doctor Availability

-The doctor will enter information into the system regarding treatments and time they can meet with patients. This will then update to the patient and receptionist view and they can then choose a time slot.

Nurse

List of Workers:



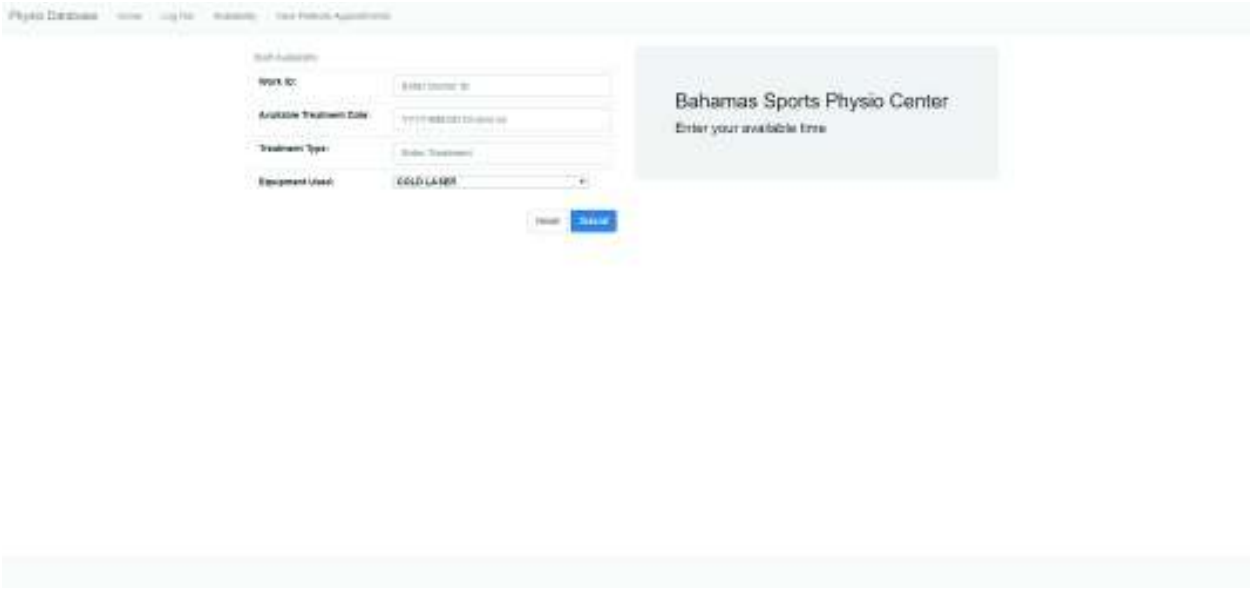
-The nurse will have limited options in this system, they will only be able to access the list of workers

Therapist

Appointment View

-The therapist will be able to access their appointments that they have with the patients

Availability Entry:



-The therapist can enter the times he or she is available for which type of treatment they will give and the equipment they will use. Once entered the patient will be able to select the newly entered time slot.

DDL Code

```
/*************************************************************************/
CREATE TABLE worker(
        ID              INT(9) PRIMARY KEY,
        F_NAME          VARCHAR(40),
        L_NAME          VARCHAR(40),
        ADDRESS         VARCHAR(255),
        PHONE           VARCHAR(20),
        CENTER          VARCHAR(255));
/*************************************************************************/
CREATE TABLE RECEPTIONIST(
        ID              INT(9) PRIMARY KEY,
        FOREIGN KEY (ID) REFERENCES worker(ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE NURSE(
        ID              INT(9) PRIMARY KEY,
        FOREIGN KEY (ID) REFERENCES worker(ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE THERAPIST(
        ID              INT(9) PRIMARY KEY,
        YEARS_EXPERIENCE        INT(3),
        FOREIGN KEY (ID) REFERENCES worker(ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE DOCTOR(
        ID              INT(9) PRIMARY KEY,
        YEARS_EXPERIENCE        INT(3),
        FOREIGN KEY (ID) REFERENCES worker(ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE TIMETABLE(
        PERIOD      DATETIME PRIMARY KEY   #RECORDS THE AVAILABLE PERIODS);
/*************************************************************************/
CREATE TABLE STAFF_INFO(
        AVAILABLE_TIME  DATETIME PRIMARY KEY,
        THERA_ID        INT(9),
        DOC_ID          INT(9),
        NURSE_ID        INT(9),
        FOREIGN KEY(AVAILABLE_TIME) REFERENCES TIMETABLE(PERIOD)
        ON DELETE CASCADE ON UPDATE CASCADE,
```

```
        FOREIGN KEY(THERA_ID) REFERENCES THERAPIST(ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY(DOC_ID) REFERENCES DOCTOR(ID)
        ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY(NURSE_ID) REFERENCES NURSE(ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE patient_record(
        P_ID            INT(9) AUTO_INCREMENT,
        APPOINTMENT_PERIOD      DATETIME,
        PRIMARY KEY(P_ID, APPOINTMENT_PERIOD),
        FOREIGN KEY(APPOINTMENT_PERIOD) REFERENCES PROVIDE(PERIOD)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE PATIENT(
        P_ID            INT(9) PRIMARY KEY AUTO_INCREMENT,
        F_NAME          VARCHAR(40),
        L_NAME          VARCHAR(40),
        ADDRESS         VARCHAR(255),
        PHONE           VARCHAR(20),
        AGE             INT(9),
        FOREIGN KEY(P_ID) REFERENCES patient_record(P_ID)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE PRESCRIPTION(
        NUMBER      INT(9) PRIMARY KEY,
        DOCTOR_DESCRIPTION      VARCHAR(255),
        DIAGNOSIS   VARCHAR(255),
        CENTER          VARCHAR(255),
        FOREIGN KEY (NUMBER) REFERENCES WRITES(NUMBER)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE APPOINTMENT(
        PRESCR_N    INT(9),
        PERIOD          DATETIME,
        PRIMARY KEY(PRESCR_N, PERIOD)
        FOREIGN KEY(PRESC_N) REFERENCES PRESCRIPTION(NUMBER)
        ON DELETE CASCADE ON UPDATE CASCADE,
        FOREIGN KEY(PERIOD) REFERENCES PROVIDE(PERIOD)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
```

```
CREATE TABLE PAYMENT(
        METHOD      ENUM("CASH","CARD","CHEQUE"),
        PERIOD      DATETIME UNIQUE,
        AMOUNT      DOUBLE(9,2),
        PRIMARY KEY(PERIOD, AMOUNT),
        FOREIGN KEY(PERIOD) REFERENCES APPOINTMENT(PERIOD)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/

CREATE TABLE CARD_AGENCY(
        NAME        VARCHAR(255)

/*************************************************************************/
CREATE TABLE TREATMENT(
        PERIOD      DATETIME,
        TYPE        VARCHAR(255),
        PRIMARY KEY(PERIOD, TYPE)
        FOREIGN KEY(PERIOD) REFERENCES TIMETABLE(PERIOD)
        ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE EQUIPMENT(
        NAME        VARCHAR(255) PRIMARY KEY);

/*********************************RELATIONS*************************************
/

CREATE TABLE STAFF_ACCESS(
R_ID                INT(9),
AVAILABLE_TIME      DATETIME,
FOREIGN KEY (R_ID) REFERENCES RECEPTIONIST (ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY(AVAILABLE_TIME) REFERENCES STAFF_INFO(AVAILABLE_TIME)
ON DELETE CASCADE ON UPDATE CASCADE);

/*************************************************************************/
CREATE TABLE REGISTER(
REGISTRATION_N      INT(9) PRIMARY KEY AUTO_INCREMENT,
R_ID                INT(9),
P_ID                INT (9),
FOREIGN KEY (R_ID) REFERENCES RECEPTIONIST (ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (P_ID) REFERENCES PATIENT(P_ID)
ON DELETE CASCADE ON UPDATE CASCADE);
```

```
/*************************************************************************/


CREATE TABLE PATIENT_ACCESS(
R_ID                INT(9),
P_ID                INT(9),
DOC_ID              INT(9),
NURSE_ID            INT(9),
FOREIGN KEY (R_ID) REFERENCES RECEPTIONIST (ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (P_ID) REFERENCES PATIENT_RECORD(P_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (DOC_ID) REFERENCES DOCTOR(ID)
ON DELETE CASCADE, ON UPDATE CASCADE,
FOREIGN KEY (NURSE_ID) REFERENCES NURSE(ID)
ON DELETE CASCADE, ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE RECEIVE(
P_ID                INT(9),
PERIOD              DATETIME,
TYPE                VARCHAR(255),
PRIMARY KEY (P_ID, PERIOD),
FOREIGN KEY (P_ID) REFERENCES PATIENT (P_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (PERIOD, TYPE) REFERENCES PROVIDE(PERIOD, TYPE)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE PROVIDE(
THERA_ID            INT(9),
PERIOD              DATETIME,
TYPE                VARCHAR(255),
PRIMARY KEY(THERA_ID, PERIOD, TYPE),
FOREIGN KEY (THERA_ID) REFERENCES THERAPIST(ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (PERIOD, TYPE) REFERENCES TREATMENT(PERIOD, TYPE)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE UPDATES(
R_ID                    INT(9),
APPOINTMENT_PERIOD      DATETIME,
PRESCR_N                INT(9),
FOREIGN KEY (R_ID) REFERENCES RECEPTIONIST (ID)
ON DELETE CASCADE ON UPDATE CASCADE,
```

```
FOREIGN KEY (APPOINTMENT_PERIOD, PRESCR_N) REFERENCES APPOINTMENT(PERIOD,
PRESCR_N)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE USES(
EQUIP_NAME          VARCHAR(255),
TREAT_PERIOD        DATETIME,
TREAT_TYPE VARCHAR(255),
PRIMARY KEY(EQUIP_NAME, TREAT_PERIOD),
FOREIGN KEY (EQUIP_NAME) REFERENCES EQUIPMENT (NAME)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (TREAT_PERIOD, TREAT_TYPE) REFERENCES TREATMENT(PERIOD, TYPE)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE AVAILABLE_AT(
PERIOD              DATETIME PRIMARY KEY,
DOC_ID              INT(9),
THERA_ID            INT(9),
NURSE_ID            INT(9),
FOREIGN KEY (PERIOD) REFERENCES TIMETABLE(PERIOD)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (DOC_ID) REFERENCES DOCTOR(ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY(THERA_ID) REFERENCES THERAPIST(ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (NURSE_ID) REFERENCES NURSE(ID)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE VERIFY(
VERIFICATION_PERIOD     ENUM(“21:00:00”) DEFAULT “21:00:00”,
CARD_AGENCY_NAME        VARCHAR(255),
PAY_AMOUNT              DOUBLE(9,2),
PAY_PERIOD              DATETIME,
PRIMARY KEY(PAY_PERIOD, PAY_AMOUNT),
FOREIGN KEY (PAY_AMOUNT,PAY_PERIOD) REFERENCES PAYMENT(AMOUNT, PERIOD)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (CARD_AGENCY_NAME) REFERENCES CARD_AGENCY(NAME)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE PAY(
P_ID                INT(9),
AMOUNT              DOUBLE(9,2),
PERIOD              DATETIME PRIMARY KEY,
```

```
FOREIGN KEY (P_ID) REFERENCES PATIENT(P_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY (AMOUNT,PERIOD) REFERENCES PAYMENT(AMOUNT,PERIOD)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE MAKE(
P_ID                    INT(9),
PRESCR_N                INT(9),
PERIOD                  DATETIME,
PRIMARY KEY(PRESCR_N),
FOREIGN KEY(P_ID) REFERENCES PATIENT(P_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY(PRESCR_N, PERIOD) REFERENCES APPOINTMENT (PRESCR_N, PERIOD)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE TAKE(
P_ID                    INT(9),
NUMBER                  INT(9) UNIQUE,
PRIMARY KEY (P_ID, NUMBER),
FOREIGN KEY (P_ID) REFERENCES PATIENT(P_ID)
ON DELETE CASCADE ON UPDATE CASCADE,
FOREIGN KEY(NUMBER) REFERENCES PRESCRIPTION(NUMBER)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
CREATE TABLE WRITES(
DOC_ID                   INT(9),
NUMBER                  INT(9) NOT NULL AUTO_INCREMENT,
PRIMARY KEY( NUMBER),
FOREIGN KEY (DOC_ID) REFERENCES DOCTOR(ID)
ON DELETE CASCADE ON UPDATE CASCADE);
/*************************************************************************/
```

Contributions(tbd)

DESIGN OF E/R DADA MODEL CONTRIBUTOR(S):;

- MAJED ALKHOURI
- MEHAL PATEL
- PETER CHEN

CONVERSION INTO RELATIONAL MODEL CONTRIBUTOR(S):

- MEHAL PATEL
- PETER CHEN

USER INTERFACE CONTRIBUTOR(S):

- ANDY NGUYEN
- MEHAL PATEL
- DINH DUC BUI
- PETER CHEN

USER GUIDE CONTRIBUTOR(S):

- MEHALPATEL

REPORTS/EXTRA REPORTS CONTRIBUTOR(S):

- MAJED ALKHOURI

DDL CODE CONTRIBUTOR(S):

- DINH DUC BUI
- MAJED ALKHOURI
- PETER CHEN

References

Front page image
http://www.cheshirephysio.co.uk/wp-content/uploads/2015/11/icon_616.png