

CPSC 471: Group 36 Final Report

Navjeet Hundal, Cody Stasyk, Victor Guan

TA: Abdelghani Guerbas

Table of Contents

1. Abstract - page 3
2. Project Design - page 4
3. EERD - page 5
4. Relational Model - page 7
5. DBMS - page 8
6. SQL statements - page 9
7. User Manual - page 18
8. Appendix - page 31

Abstract

A local holistic health clinic, Healthy By Nature, is unhappy with its online appointment booking system: it is an expensive monthly cost, and it is hosted on a separate, unreliable server. Our project aimed to replace this system with our own: a free, locally hosted appointment booking system that would alleviate the clinic's two major pain points.

Our project succeeded on both of those central points: we managed to design an online booking system that is hosted on the same server as this clinic's website, and our booking solution is provided to them free of charge.

Our project managed to replicate most of the core features of their current solution, though not all of them. We estimate that only ~90% of the target features made it into this design that we are submitting right now. The appointment-approving mechanism for the staff with administrative permission is the most prominent feature that did not make it into this final design and that will have to be implemented at a later date for the clinic.

PROJECT DESIGN

Users

There are two main different types of users of our appointment booking system: **customers** and **staff members**.

The customers are ultimately who this system is designed for: it allows them to select an appointment type, select the therapist they want to see, and then make an appointment request.

The staff members manage the system and maintain the information in the system. They log-in and access the staff-only back end of the booking system and can do things like add therapists to the system, add services to the system, associate services with therapists, view their appointments, book time off (for vacations), look up customers, etc....

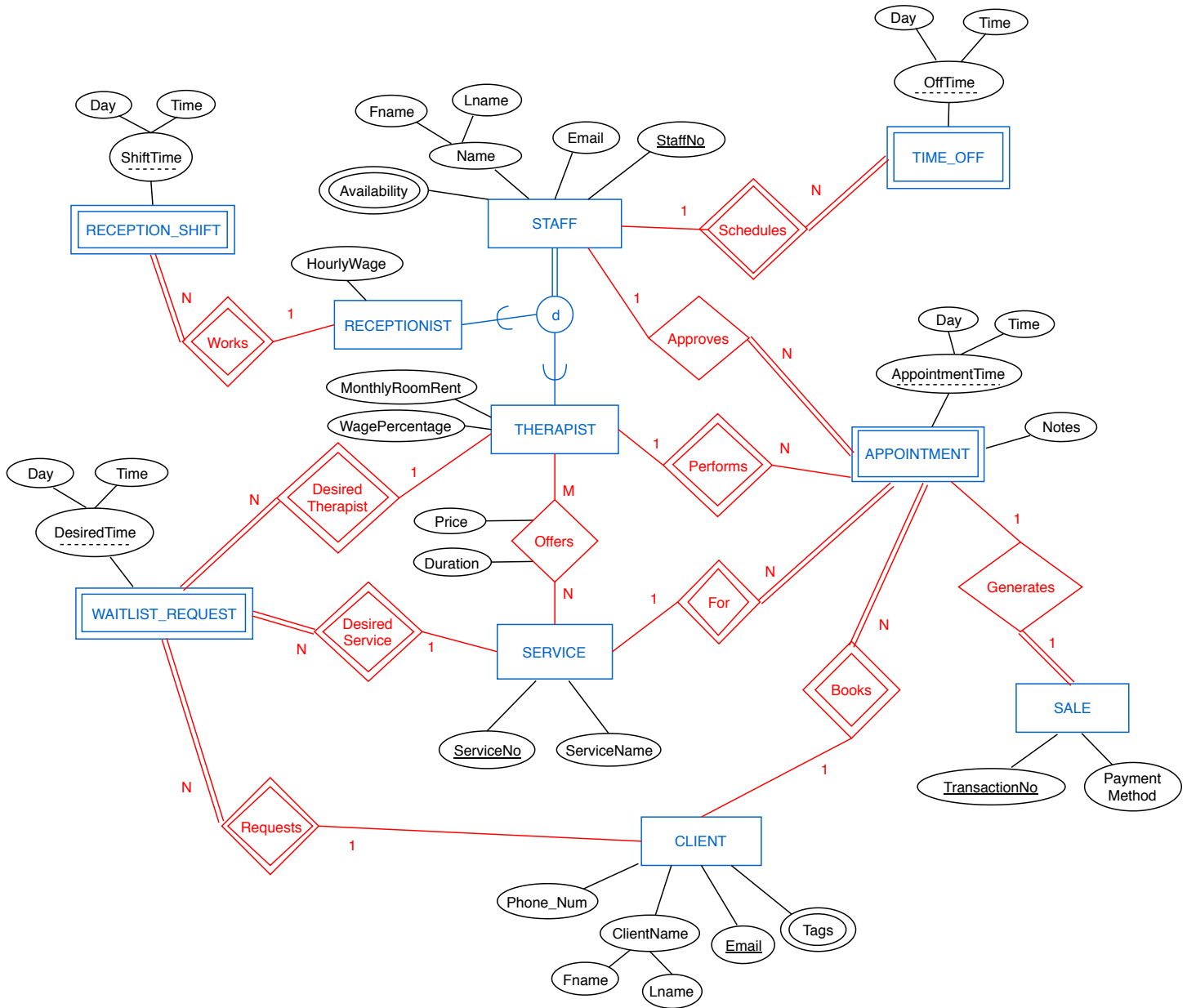
Customer Functionality: customer users are able to...

- View the different therapy types and select one
- View different therapists that offer the selected service and pick one
- Make an appointment request on a specific day and time for that service, with that therapist

Staff Functionality: staff users are able to...

- Manage therapists: view all, add new, update existing, and remove existing ones
- Manage receptionists: (same functions)
- Manage reception shifts: (same functions)
- Manage the services offered at the clinic: (same functions)
- Manage the services offered by specific therapists: (same functions)
- Manage time off for staff: (same functions)
- Manage customer information: (same functions)
- Manage appointments: (same functions)
- Manage waitlist requests: (same functions)
- View sales data
- View all staff in the system

EERD



Assumptions:

The decisions made in the design of this EERD are not based on assumptions as much they are based on the specifications that were communicated by the owners of the business we are making this database application for.

Effort has been made to name all of the entity types and their attributes in an obvious, self-documenting way, but the cardinality and participation of the entity types in the various relationships probably deserve explanation:

A STAFF member can (but does not have to) schedule any number of chunks of TIME_OFF, but each scheduled chunk of TIME_OFF will (and must) apply to only one STAFF member.

A RECEPTIONIST can (but doesn't have to) work any number of RECEPTION_SHIFTS, and each RECEPTION_SHIFT is always scheduled for exactly one RECEPTIONIST.

A CLIENT may (but does not have to) request any number of WAITLIST_REQUESTs, and each WAITLIST_REQUEST is always made for a single CLIENT. The WAITLIST_REQUEST is made to specify exactly one desired SERVICE and one THERAPIST, and it is possible for those THERAPISTs and SERVICES to appear in many different WAITLIST_REQUESTs.

Each THERAPIST typically offers many SERVICES, but they don't have to: this database will be functioning as part of an appointment booking system, and so when a THERAPIST goes on an extended leave of absence (maternity leave, for example), the database will be modified to remove all SERVICES offered by that THERAPIST. The THERAPIST will still be in the database, but they will not be offering any SERVICES during that time.

Each SERVICE can be offered by many THERAPISTs, but it doesn't have to be: an example would be in that last example with the THERAPIST on maternity leave if she were the only one offering that SERVICE. The SERVICE she offers would still be in the database, but nobody would be offering it for a period of time in her absence.

APPOINTMENTS are booked by (and for) one CLIENT, and a CLIENT can book as many appointments as they want.

Each APPOINTMENT also involves exactly one SERVICE and one THERAPIST, while those SERVICES and THERAPISTs can feature in many appointments.

APPOINTMENTS also require approval by a STAFF member in order to be considered a completely booked appointment. After a CLIENT books an APPOINTMENT, exactly one STAFF member has to approve the APPOINTMENT, and a STAFF member can approve many appointments.

And APPOINTMENTS can generate a SALE, but not all of them do: sometimes CLIENTs miss their APPOINTMENT or call and cancel it at the last minute. In those cases, the APPOINTMENT will remain in the database, but no SALE will have been generated. Of course, if a SALE is made, then it will be for exactly one APPOINTMENT.

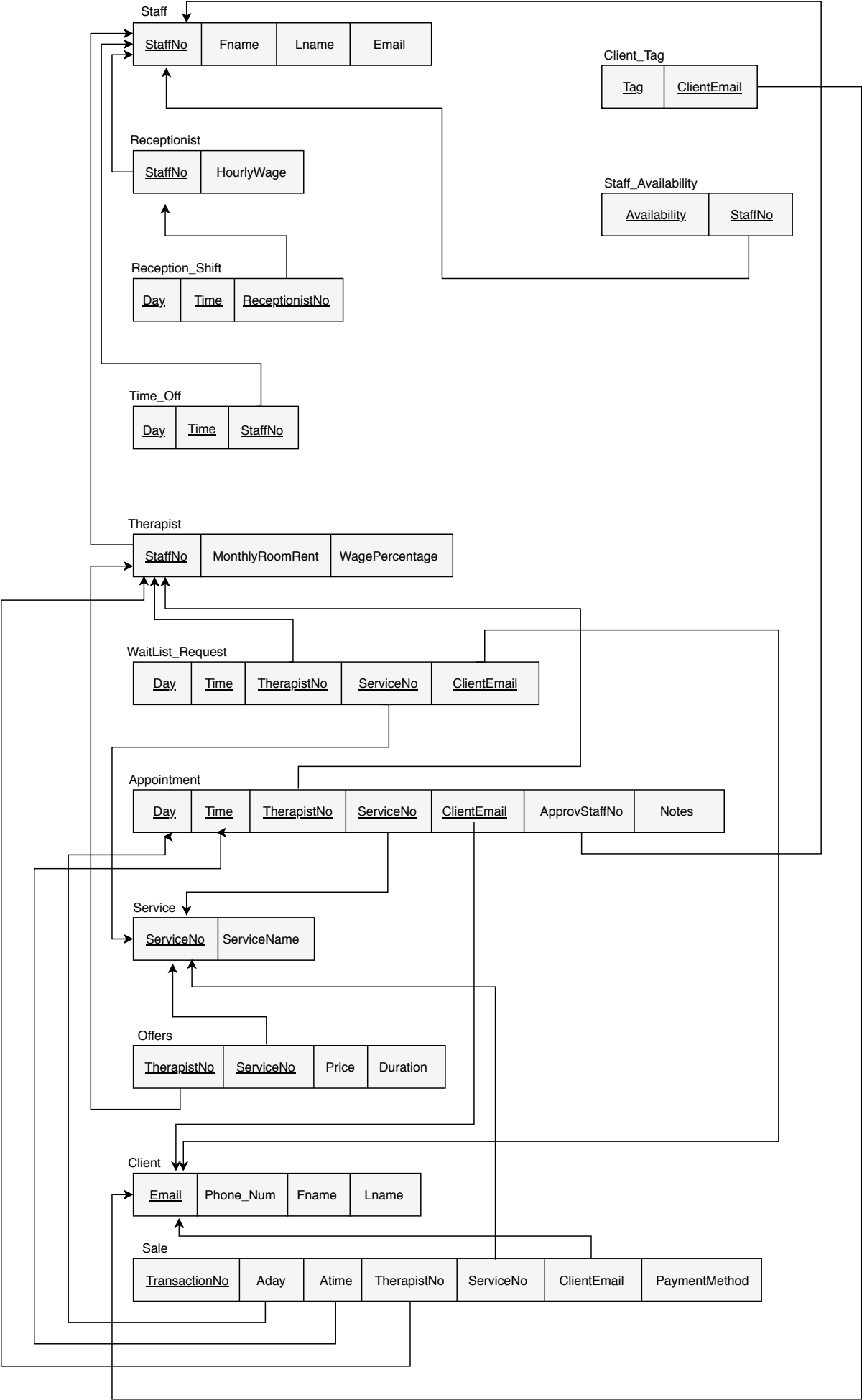
IMPLEMENTATION

Relational Model

We followed the standard algorithm for converting an EERD into a relational schema to derive our clinic booking system schema diagram.

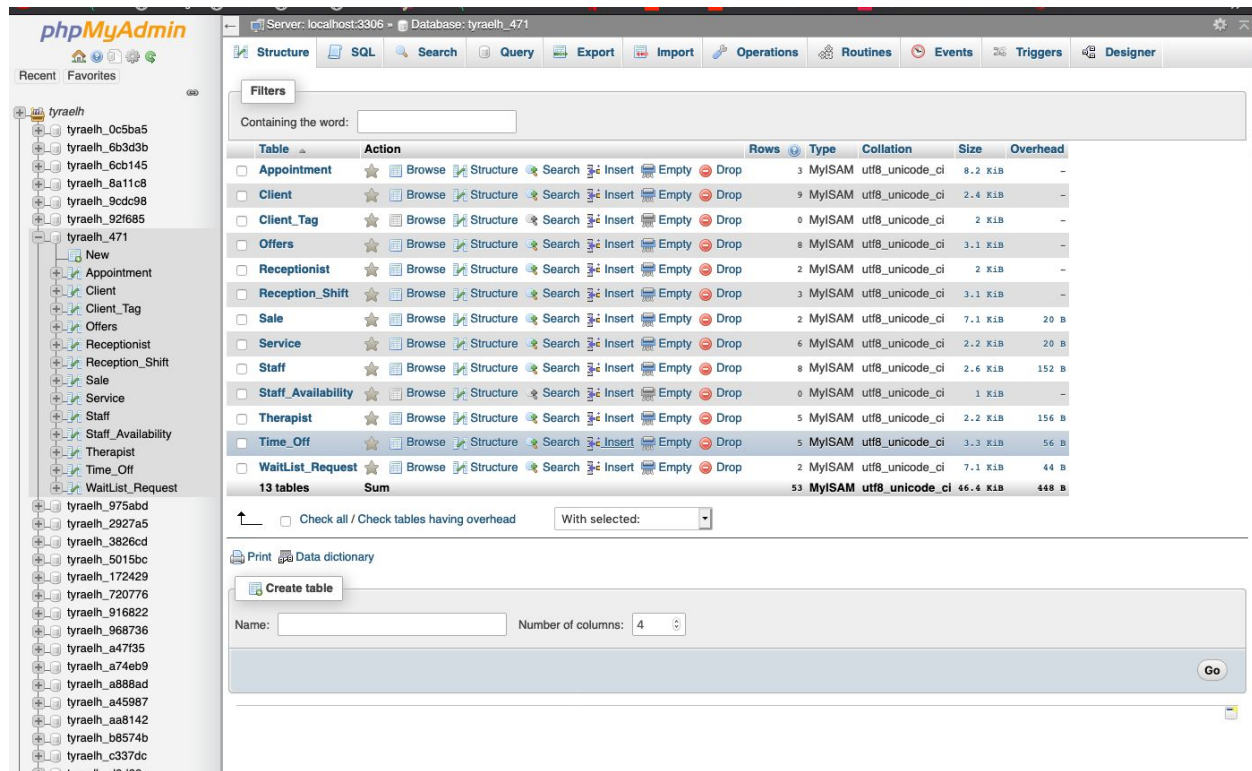
On the next page is the diagram of our relational schema.

Relational Model



DBMS

We used the phpMyAdmin database management system to assist us in managing our MySQL database for the project:



SQL statements used to create our database

(You can find these statements in the included zipped files, in “create-entire-schema.php”)

```
CREATE TABLE Staff (  
  StaffNo INT NOT NULL PRIMARY KEY,  
  Fname VARCHAR(30),  
  Lname VARCHAR(30),  
  Email VARCHAR(50),  
  Password VARCHAR(50) )
```

```
CREATE TABLE Therapist (  
  StaffNo INT,  
  MonthlyRoomRent INT,  
  WagePercentage INT,  
  FOREIGN KEY (StaffNo) REFERENCES Staff(StaffNo))
```

```
CREATE TABLE Receptionist (  
  StaffNo INT,  
  HourlyWage INT,  
  FOREIGN KEY (StaffNo) REFERENCES Staff(StaffNo) )
```

```
CREATE TABLE Client (  
  Email varchar(50) NOT NULL PRIMARY KEY,  
  Phone_Num varchar(20),  
  Fname varchar(30),  
  Lname varchar(30) )
```

```
CREATE TABLE Service (  
  ServiceNo int NOT NULL PRIMARY KEY,  
  ServiceName varchar(50) )
```

```
CREATE TABLE Reception_Shift (  
  Day varchar(30) NOT NULL,  
  Time varchar(30) NOT NULL,  
  ReceptionistNo int,  
  PRIMARY KEY (Day, Time, ReceptionistNo),  
  FOREIGN KEY (ReceptionistNo) REFERENCES Receptionist(StaffNo) )
```

```
CREATE TABLE Appointment (  
  Day varchar(30) NOT NULL,
```

Time varchar(30) NOT NULL,
 TherapistNo int,
 ServiceNo int,
 ClientEmail varchar(50),
 ApproveStaffNo int,
 Notes varchar(255),
 PRIMARY KEY (Day, Time, TherapistNo, ServiceNo, ClientEmail),
 FOREIGN KEY (TherapistNo) REFERENCES Therapist(StaffNo),
 FOREIGN KEY (ServiceNo) REFERENCES Service(ServiceNo),
 FOREIGN KEY (ClientEmail) REFERENCES Client(Email),
 FOREIGN KEY (ApproveStaffNo) REFERENCES Staff(StaffNo))

CREATE TABLE Time_Off (
 Day varchar(30) NOT NULL,
 Time varchar(30) NOT NULL,
 StaffNo int,
 PRIMARY KEY (Day, Time, StaffNo),
 FOREIGN KEY (StaffNo) REFERENCES Staff(StaffNo))

CREATE TABLE WaitList_Request (
 Day varchar(30) NOT NULL,
 Time varchar(30) NOT NULL,
 TherapistNo int,
 ServiceNo int,
 ClientEmail varchar(50),
 PRIMARY KEY (Day, Time, TherapistNo, ServiceNo, ClientEmail),
 FOREIGN KEY (TherapistNo) REFERENCES Therapist(StaffNo),
 FOREIGN KEY (ServiceNo) REFERENCES Service(ServiceNo),
 FOREIGN KEY (ClientEmail) REFERENCES Client(Email))

CREATE TABLE Offers (
 TherapistNo int,
 ServiceNo int,
 Price int,
 Duration int,
 PRIMARY KEY (TherapistNo, ServiceNo),
 FOREIGN KEY (TherapistNo) REFERENCES Therapist(StaffNo),
 FOREIGN KEY (ServiceNo) REFERENCES Service(ServiceNo))

CREATE TABLE Sale (
 TransactionNo int NOT NULL PRIMARY KEY,
 Aday varchar(30) NOT NULL,
 Atime varchar(30) NOT NULL,

```
TherapistNo int,  
ServiceNo int,  
ClientEmail varchar(50),  
PaymentMethod varchar(30),  
FOREIGN KEY (Aday) REFERENCES Appointment(Day),  
FOREIGN KEY (Atime) REFERENCES Appointment(Time),  
FOREIGN KEY (TherapistNo) REFERENCES Therapist(StaffNo),  
FOREIGN KEY (ServiceNo) REFERENCES Service(ServiceNo),  
FOREIGN KEY (ClientEmail) REFERENCES Client(Email) )
```

```
CREATE TABLE Client_Tag (  
Tag varchar(50),  
ClientEmail varchar(50),  
PRIMARY KEY (Tag, ClientEmail),  
FOREIGN KEY (ClientEmail) REFERENCES Client(Email) )
```

```
CREATE TABLE Staff_Availability (  
Availability varchar(50),  
StaffNo int,  
PRIMARY KEY (Availability, StaffNo),  
FOREIGN KEY (StaffNo) REFERENCES Staff(StaffNo) )
```

Additional SQL statements used:

To protect against SQL injection attacks, many of our statements are “prepared” (ie: parameterized):

```
// add a new Client
$statement = $con->prepare("INSERT INTO Client VALUES (?, ?, ?, ?)");
$statement->bind_param("ssss", $ClientEmail, $Phone_Num, $Fname, $Lname);
$statement->execute();

// add the Appointment
$statement = $con->prepare("INSERT INTO Appointment VALUES (?, ?, ?, ?, ?, -1, ?)");
$statement->bind_param("ssiiss", $Day, $Time, $TherapistNo, $ServiceNo, $ClientEmail, $Notes);
$statement->execute();
```

Other times, this is not necessary:

```
// show all appointments that Staff member has
$result = mysqli_query($con, "SELECT * FROM Appointment WHERE TherapistNo = " . $TherapistNo);
```

Regardless, here are the rest of our queries, mixed in with PHP:

```
// --- Staff SELECT -----
$statement = $con->prepare("SELECT * FROM Staff WHERE StaffNo = ?");
$statement->bind_param("i", $TherapistNo);
$statement->execute();

// Select all staff who offer a particular service
$result = mysqli_query($con, "SELECT * FROM Offers, Staff WHERE Offers.ServiceNo = " .
$ServiceNo . " AND Staff.StaffNo = Offers.TherapistNo");
```

```

// --- Offers SELECT -----
$statement = $con->prepare("SELECT * FROM Offers WHERE TherapistNo = ?");
$statement->bind_param("i", $StaffNo);
$statement->execute();

// --- Service SELECT -----
$statement = $con->prepare("SELECT * FROM Service WHERE ServiceNo = ?");
$statement->bind_param("i", $ServiceNo);
$statement->execute();

// view all services offered by a particular Staff member
$result = mysqli_query($con, "SELECT * FROM Staff, Service, Offers WHERE Staff.StaffNo = " .
$StaffNo . " AND Staff.StaffNo = Offers.TherapistNo AND Offers.ServiceNo =
Service.ServiceNo");

// add a new service offered by a therapist
$statement = $con->prepare("INSERT INTO Offers VALUES (?, ?, ?, ?)");
$statement->bind_param("iiii", $TherapistNo, $ServiceNo, $Price, $Duration);
$statement->execute();

// update a service offered by a therapist
$statement = $con->prepare("UPDATE Offers SET TherapistNo = ?, ServiceNo = ?, Price = ?,
Duration = ? WHERE TherapistNo = ? AND ServiceNo = ?");
$statement->bind_param("iiiiii", $TherapistNo, $ServiceNo, $Price, $Duration, $TherapistNo,
$ServiceNo);
$statement->execute();

// delete a service offered by a therapist
$statement = $con->prepare("DELETE FROM Offers WHERE TherapistNo = ? AND ServiceNo =
?");
$statement->bind_param("ii", $TherapistNo, $ServiceNo);
$statement->execute();

```

```
// add a sale
$statement = $con->prepare("INSERT INTO Sale VALUES (?, ?, ?, ?, ?, ?)");
$statement->bind_param("issiiiss", $TransactionNo, $Aday, $Atime, $TherapistNo, $ServiceNo,
$ClientEmail, $PaymentMethod);
$statement->execute();
```

```
// update a sale
$statement = $con->prepare("UPDATE Sale SET TransactionNo = ?, Aday = ?, Atime = ?,
TherapistNo = ? , ServiceNo = ?, ClientEmail = ?, PaymentMethod = ? WHERE TransactionNo =
?");
$statement->bind_param("issiiiss", $TransactionNo, $Aday, $Atime, $TherapistNo, $ServiceNo,
$ClientEmail, $PaymentMethod, $TransactionNo);
$statement->execute();
```

```
// delete a sale
$statement = $con->prepare("DELETE FROM Sale WHERE TransactionNo = ?");
$statement->bind_param("i", $TransactionNo);
$statement->execute();
```

```
// select all Receptionists and join them with their Staff information
$result = mysqli_query($con, "SELECT * FROM Receptionist, Staff WHERE Staff.StaffNo =
Receptionist.StaffNo");
```

```
// select all shifts for a particular Receptionist
$statement = $con->prepare("SELECT * FROM Reception_Shift WHERE ReceptionistNo = ?");
$statement->bind_param("i", $StaffNo);
$statement->execute();
```

```
// add a receptionist shift
$statement = $con->prepare("INSERT INTO Reception_Shift VALUES (?, ?, ?)");
$statement->bind_param("ssi", $Day, $Time, $ReceptionistNo);
$statement->execute();
```

```
// update a receptionist shift
$statement = $con->prepare("UPDATE Reception_Shift SET Day = ?, Time = ?, ReceptionistNo =
? WHERE Day = ? AND Time = ? AND ReceptionistNo = ?;");
$statement->bind_param("ssissi", $Day, $Time, $ReceptionistNo, $PreviousDay,
$PreviousTime, $ReceptionistNo);
$statement->execute();
```

```
// delete a receptionist shift
$statement = $con->prepare("DELETE FROM Reception_Shift WHERE Day = ? AND Time = ?
AND ReceptionistNo = ?");
$statement->bind_param("ssi", $Day, $Time, $ReceptionistNo);
$statement->execute();
```

```
// select all receptionist shifts and extended data connected with the receptionists (join)
$result = mysqli_query($con, "SELECT * FROM Receptionist, Staff, Reception_Shift WHERE
Staff.StaffNo = Receptionist.StaffNo AND Reception_Shift.ReceptionistNo = Staff.StaffNo");
```

```
// select a particular sale
$statement = $con->prepare("SELECT * FROM Sale WHERE TransactionNo = ?");
$statement->bind_param("i", $TransactionNo);
$statement->execute();
```

```
// add a service
$statement = $con->prepare("INSERT INTO Service VALUES (?,?)");
$statement->bind_param("is", $ServiceNo, $ServiceName);
$statement->execute();
```

```
// update a service
$statement = $con->prepare("UPDATE Service SET ServiceName = ? WHERE ServiceNo = ?");
$statement->bind_param("si", $ServiceName, $ServiceNo);
$statement->execute();
```

```

// delete a service
$statement = $con->prepare("DELETE FROM Service WHERE ServiceNo = ?");
$statement->bind_param("i", $ServiceNo);
$statement->execute();

// add a therapist
$statement = $con->prepare("INSERT INTO Staff VALUES (?, ?, ?, ?, ?)");
$statement->bind_param("issss", $StaffNo, $Fname, $Lname, $Email, $Password);
$statement->execute();
$statement2 = $con->prepare("INSERT INTO Therapist VALUES (?, ?, ?)");
$statement2->bind_param("iii", $StaffNo, $MonthlyRoomRent, $WagePercentage);
$statement2->execute();

// update a therapist
$statement = $con->prepare("UPDATE Staff SET Fname = ?, Lname = ?, Email = ?, Password = ?
WHERE StaffNo = ?");
$statement->bind_param("ssssi", $Fname, $Lname, $Email, $Password, $StaffNo);
$statement->execute();
$statement = $con->prepare("UPDATE Therapist SET MonthlyRoomRent = ?, WagePercentage =
? WHERE StaffNo = ?");
$statement->bind_param("iii", $MonthlyRoomRent, $WagePercentage, $StaffNo);
$statement->execute();

// delete a therapist
$statement = $con->prepare("DELETE FROM Therapist WHERE StaffNo = ?");
$statement->bind_param("i", $StaffNo);
$statement->execute();
$statement = $con->prepare("DELETE FROM Staff WHERE StaffNo = ?");
$statement->bind_param("i", $StaffNo);
$statement->execute();

// add time off
$statement = $con->prepare("INSERT INTO Time_Off VALUES (?, ?, ?)");
$statement->bind_param("ssi", $Day, $Time, $StaffNo);
$statement->execute();

```



```
// update time off
$statement = $con->prepare("UPDATE Time_Off SET Day = ?, Time = ? WHERE StaffNo = ?");
$statement->bind_param("ssi", $Day, $Time, $StaffNo);
$statement->execute();
```

```
// delete time off
$statement = $con->prepare("DELETE FROM Time_Off WHERE StaffNo = ? AND Day = ? AND
Time = ?");
$statement->bind_param("iss", $StaffNo, $Day, $Time);
$statement->execute();
```

User Manual

Link to the test website where we hosted our project, hosted on the same IP address/server as the Clinic:

<https://www.iwanttowalkagain.com/user471/>

Home page:



Welcome to Healthy By Nature's Appointment Booking System!

[Book an Appointment](#)

[Staff Login](#)

This is the homepage where a client can book an appointment or a staff member can login

Book an appointment page:

What type of appointment would you like to book?

Click on the service you are interested in...
Osteopathy - First Appointment
Osteopathy - Follow Up
Naturopathy - First Appointment
Naturopathy - Follow Up
Acupuncture - First Appointment
Acupuncture - Follow Up

[Go Back](#)

This is the page when a client presses the book an appointment link. Here they can choose a service that they desire

These are the therapists who offer that service.
Click the one you want to book with:

Therapist
Antoine Michelle
Sandra May

[Go Back](#)

This is the page after the client has chosen the service that they want.

This therapist is currently booked (unavailable during these times):

Day	Time
Feb 28	1pm
DD-MM-YYYY	X AM/PM

You are booking a Osteopathy - First Appointment appointment with Antoine Michelle
Please complete the details for this appointment, below:

Desired Appointment Day:

Desired Appointment Time:

Your First Name:

Your Last Name:

Your email:

Your phone number:

Any notes or requests?:

Once the client has picked a therapist they fill in their information to book an appointment.

Staff login page:



Username

Password

When you click the link to staff login from the home page it takes you to this login page



Failed Login, please try again

Try Again

If the login is wrong the user gets directed to this page where they can press try again button to go back to the login screen.

Staff home page:



Welcome to the booking system's back end...

Staff Options:

[Staff](#)
[Services](#)
[Therapists](#)
[Therapist Services](#)
[Receptionists](#)
[Receptionist Shifts](#)
[Time-Off](#)
[Sales](#)
[Client](#)
[WaitList](#)
[Appointment](#)

Logout

Home Page

This is the staff home page once the staff has logged in

Staff view page:

Staff members:

Staff Number	First Name	Last Name	Email	update?	delete?
8888	Nancy	Cookies	nancy@hbn.ca	Update	DELETE
99	Sandra	May	smaybee@gmail.com	Update	DELETE
15	Stephanie	Tsun	steph@hbn.ca	Update	DELETE
13	tim	rip	1@hotmail.com	Update	DELETE
9998	Antoine	Michelle	antoine@hbn.ca	Update	DELETE
1	God	Admin	1	Update	DELETE
9999	James	McSchmurgelton	james91@hbn.ca	Update	DELETE
3121	Amber	Franzenstein	amber21@yahoo.com	Update	DELETE

[Add a new staff member](#)

[Go Back](#)

When you click the staff link it takes you to the staff view page where you can add a new staff member or update a staff member. You can also delete a staff member if they have left the company.

Staff update page:

Update staff number: 8888

First Name:

Last Name:

E-mail:

Account Password:

[update](#)

When you press the update link on the staff page it takes you to this update page where you can update a staff members information.

Staff Delete :

Staff members:

Staff Number	First Name	Last Name	Email	update?	delete?
8888	Nancy	Cookies	nancy@hbn.ca	Update	DELETE
99					DELETE
15					DELETE
13					DELETE
9998					DELETE
1	God	Admin	1	Update	DELETE
9999	James	McSchmurgelton	james91@hbn.ca	Update	DELETE
3121	Amber	Franzenstein	amber21@yahoo.com	Update	DELETE

Do you want to delete this staff member?

[Cancel](#) [OK](#)

[Add a new staff member](#)

[Go Back](#)

This is what happens when you want press the delete option in the staff view page. It gives you the option to cancel or delete the staff member. Also all the delete links do the same thing and all look alike.

Staff add page:

Add a new staff member:

Staff Number:

First Name:

Last Name:

E-mail:

Account Password:

When you press add staff member link on the staff page it takes you here where you can add a new staff member.

Service view page:

Services Offered:

Service Name	update?	delete?
Osteopathy - First Appointment	Update	DELETE
Osteopathy - Follow Up	Update	DELETE
Naturopathy - First Appointment	Update	DELETE
Naturopathy - Follow Up	Update	DELETE
Acupuncture - First Appointment	Update	DELETE
Acupuncture - Follow Up	Update	DELETE

[Add a new service](#)

[Go Back](#)

When you press the service link on the staff home page it takes you to the service view page.

Service update page:

Update service: Osteopathy - First Appointment

Service Name:

[update](#)

When you press update on the service view page it takes you to the update page where you can update a service.

Service add page:

Add a new service:

Service Number:

Service Name:

[add](#)

When you press add service on the service view page it takes you here where a staff member can add a new service.

Therapist view page:

Here's all our Therapists:

Staff Number	First Name	Last Name	Email	Monthly Room Rent	Wage Percentage	update?	delete?
15	Stephanie	Tsun	steph@hbn.ca	2200	80	Update	DELETE
99	Sandra	May	smaybee@gmail.com	1700	65	Update	DELETE
9998	Antoine	Michelle	antoine@hbn.ca	2400	70	Update	DELETE
9999	James	McSchmurgelton	james91@hbn.ca	1900	60	Update	DELETE

[Add a new therapist member](#)

[Go Back](#)

When the staff member presses the therapist link it takes them to the view therapist page.

Therapist update page:

Update therapist number: 15

First Name:

Last Name:

E-mail:

Account Password:

Monthly Room Rent:

Wage Percentage:

[update](#)

When you use the update link on the therapist view page it takes you to the page to update a therapist.

Therapist service page:

Click a Therapist to view/modify the services they offer:

Therapist
Stephanie Tsun
Sandra May
Antoine Michelle
James McSchmurgelton

[Go Back](#)

When you press the therapist service link on the staff page it takes you to the therapist service page.

Therapist service view/modify page:

Services offered by Stephanie Tsun:

Service	Duration	Price	Update?	Delete?
Naturopathy - First Appointment	90 min	\$220	Update	Delete
Naturopathy - Follow Up	45 min	\$180	Update	Delete

[Add a new service for Stephanie Tsun](#)

[Go Back](#)

When you press a therapist from the therapist service page it takes you here where you can update or add a new service for the therapist that you clicked on from the last page.

Receptionist view page:

Here's all our Receptionist:

Staff Number	First Name	Last Name	Email	Hourly Wage	update?	delete?
8888	Nancy	Cookies	nancy@hbn.ca	80	Update	DELETE
3121	Amber	Franzenstein	amber21@yahoo.com	0	Update	DELETE

[Add a new receptionist member](#)

[Go Back](#)

When you press the receptionist link it takes you to the receptionist view page. Here you can add a new receptionist or update/ delete one.

Receptionist shifts view page:

Receptionist Shifts:

Receptionist	Shift Day	Shift Time	update?	delete?
Amber Franzenstein	Jan 10 2020	9am-5pm	Update	DELETE
Amber Franzenstein	Jan 12 2020	noon - 8pm	Update	DELETE
Nancy Cookies	Jan 8 2020	all day	Update	DELETE

[Add a new reception shift](#)

[Go Back](#)

When you press the receptionist shift link from the staff home page it takes you here where you can add a new shift, or update a current receptionist shift.

Time off view page:

Current User(Session): 1

Current User Number(Session): 1

All Time_Offs:

Staff Number	First Name	Last Name	Day	Time	update?	delete?
9999	James	McSchmurgelton	Dec 11 2019	9:00	Update	DELETE

[Add a new Time-Off](#)

[Go Back](#)

When you press the time off link from the staff home page it takes you to the time off view page where staff can book their days off.

Sales view page:

Here's all Sales:

Transaction Number	Day	Time	Therapist Number	Service Number	Client Email	Payment Method	update?	delete?
54	Nov 20 2019	10:30am	9998	11	john45@hotmail.com	Rupees1	Update	DELETE

[Add a new sale](#)

[Go Back](#)

When you press the sales link from the staff home page it will take you to the sales view page where staff can view the sales,update them and add new sales.

Client information view page:

Here's all our Clients:

Client Email/th>	Phone Number/th>	First Name	Last Name	update?	delete?
monkey@fthis.com	403-768-3821	Jalal	Teacherson	Update	DELETE
john45@hotmail.com	587-234-5932	John	Smithers	Update	DELETE
Alt@Esp.com	403-999-9999	Alt	Esp	Update	DELETE
AltSon@email.com	(403)456-6545	Son	Esp	Update	DELETE
del	19283109	alsmo	topsd	Update	DELETE
please@gmail.com	290-208-1123	Be	Gone	Update	DELETE
cindy@gogo.com	302-234-8421	Cindy	Li	Update	DELETE
sl@godaddy.com	587-234-2381	Sandra	Loop	Update	DELETE
123@hotmail.com	123-445-234	Alt	Esp	Update	DELETE

Add a new client member

[Go Back](#)

When you press the client link from the staff home page it takes you to the client information view page where the staff can update client information/ add a new client or delete a client.

Waitlist view page:

Waitlists:

Day	Time	Therapist No.	Service No.	Email	update?	delete?
Dec 10 2000	10:00	15	1	asdf@email.com	Update	DELETE
Feb 20 2020	8:00PM	15	2	Alt@Esp.com	Update	DELETE

Add a new WaitList

[Go Back](#)

When you press the waitlist link from the staff home page it takes you to the waitlist view page where staff can view the people in a waitlist and update it.

Appointment view page:

Here's all appointments:

Day	Time	Therapist Number	Service Number	Client Email	Appproved Staff Number	Notes	update?	delete?
Feb 28	1pm	9998	1	Mary@basic.com	-1		Update	DELETE
01-01-2020	10 AM	99	1	sl@godaddy.com	-1	nope :)	Update	DELETE
DD-MM-YYYY	X AM/PM	9998	1	123@hotmail.com	-1		Update	DELETE

[Add a new appointment](#)

[Go Back](#)

When you press appointment link on the staff page it takes you to the appointment view page where staff can see the appointments of clients and they can make new appointments/ update existing ones or cancel an appointment.

Appendix: Sample Data Tuples

Here are some instances of sample data in our database:

Staff

StaffNo	Fname	Lname	Email	Password
8888	Nancy	Cookies	nancy@hbn.ca	...
99	Sandra	May	smaybee@gmail.com	1
15	Stephanie	Tsun	steph@hbn.ca	123
13	tim	rip	1@hotmail.com	1
9998	Antoine	Michelle	antoine@hbn.ca	antoine123
1	God	Admin	1	1
9999	James	McSchmurgelton	james91@hbn.ca	james@123
3121	Amber	Franzenstein	amber21@yahoo.com	1

Therapist

StaffNo	MonthlyRoomRent	WagePercentage
15	2200	80
99	1700	65
9998	2400	70
9999	1900	60

Receptionist

StaffNo	HourlyWage
8888	80
3121	0

Reception_Shift

Day	Time	ReceptionistNo
23-06-2020	7 AM	8888
Jan 11 2020	9am-5pm	3121
Jan 12 2020	noon - 8pm	3121
Jan 8 2020	all day	8888

Time_Off

Day	Time	StaffNo
Dec 10 2000	12:00	3
Dec 10 2000	20:00	3
Dec 11 2019	9:00	9999
Jan 10 2019	12:00	2
Jan 10 2019	20:00	535

Service

ServiceNo	ServiceName
1	Osteopathy - First Appointment
2	Osteopathy - Follow Up
11	Naturopathy - First Appointment
12	Naturopathy - Follow Up
21	Acupuncture - First Appointment
22	Acupuncture - Follow Up

Offers

TherapistNo	ServiceNo	Price	Duration
9998	2	122	40
9998	1	145	65
9999	22	90	35
99	1	140	70
99	2	110	45
9999	21	170	80
15	11	220	90
15	12	180	45

Client

Email	Phone_Num	Fname	Lname
monkey@fthis.com	403-768-3821	Sampson	Teachberg
john45@hotmail.com	587-234-5932	John	Smithers
Alt@Esp.com	403-999-9999	Alt	Esp
AltSon@email.com	(403)456-6545	Son	Esp
del	19283109	Fred	Gallagher
please@gmail.com	290-208-1123	No	More
cindy@gogo.com	302-234-8421	Cindy	Li
sl@godaddy.com	587-234-2381	Sandra	Loop
123@hotmail.com	123-445-234	Chao	Zhehan

Appointment

Day	Time	TherapistNo	ServiceNo	ClientEmail	ApprovStaffNo	Notes
Feb 28	1pm	9998	1	Mary@basic.com		
01-01-2020	10 AM	99	1	sl@godaddy.com		nope :)

WaitList_Request

Day	Time	TherapistNo	ServiceNo	ClientEmail
Dec 10 2000	10:00	15	1	asdf@email.com
Feb 20 2020	8:00PM	15	2	Alt@Esp.com