Assignment 1

Name: Abhisha Thaker

Banner number: B00937694

PROBLEM 1

Follow all the steps in the given order. For you, some steps could be optional and depends on the solution you create.

Hypothetical Problem Scenario: A new business "Hello12 Management team" is trying to establish a "Bed & Breakfast" in Halifax region. They need your help in building their backend information system. They are new in Nova Scotia, and do not have much knowledge about the surroundings or policies etc. They just gave you the following abstract requirements and ideas. "

•Visit multiple bed and breakfast or hotel websites and understand the concepts

•Once you get the business domain knowledge start building a conceptual design for the team

•We can only tell you that for our Bed & Breakfast, we want to offer different types of breakfast items, there will be different types of rooms, and we do not have a swimming pool or gymnasium. However, we can give coupons for those, so that boarders can access those facilities outside. – Hello12 Management team"

Step 1. Identify at least 8 websites related to hotels or lodging facilities, which are useful for the given requirements, and create a table with a column containing websites URLs, and another column highlighting what information you gathered from those websites. You do not have to run any automated crawler or web scrapping tool. This is just a manual process, so that you get the business domain knowledge.

No.	Website URL	Information Gathered
1.	https://www.marriott.com/en-us/hotels/yhzwi-the-westin-nova-scotian/overview/?scid=f2ae0541-1279-4f24-b197-a979c79310b0	Rooms – Availability of room, Rate, minimum person per room, occupancy available, name of the room,

	https://www.seaportsocialhalifax.com/our-menus	room_id, checkin date, checkout date. User – Address, City, State, Zip Code, Planner or a Travel Agent. Table – no of tables, slot, time, day, seating_type Usertable – Firstname, lastname, phone number, email, occasion, any special requests. Menu – TypeofMenu, price, menuname, menuid, phone_number.
2.	https://www.hilton.com/en/hotels/yhzhhdt-the-hollis-halifax/	Rooms – typeofroom Facilities - facility_id, facilityname, roomid,
3.	https://lordnelsonhotel.ca/	Hotel – Hotelid, hotel name, hotel phone number, hotel contactnumber, email address, website URL
4.	https://www.princegeorgehotel.com/	There can be multiple types of rooms such as - Deluxe guest room, crown service room,

		Deluxe king suits, Executive suits, The prince suite. But 1 room can be only of one type.
5.	https://www.cambridgesuiteshalifax.com/	Food – onsite service time, offsite service time, menu_image, quantity, price.
6.	https://ca.hotels.com/ho196368/four-points-by-sheraton-halifax-halifax-canada/	Other policies – pet policy, parking policy Location – geocoordinates can be considered as derived attribute
7.	https://www.digbypines.ca/dining-at-digby-pines https://www.digbypines.ca/gift-cards https://www.digbypines.ca/about-digby-pines	Dining on Terrace Different type of packages information. Offers and coupons can be given. Photos of menu and photos of rooms.
8.	https://www.foxharbr.com/	Roomservice information, Room amenities information and viewing menu and order

Step 2. Identify at least 9 unique entities (sub-types are not counted as separate entities for this problem; Only the supertype will be considered as 1 entity), which

must contain at least 1 weak entity. Write your assumption in a short paragraph explaining why you considered these entities, and how do you justify your strong entities as strong and weak entities as weak.

9 unique entities are as follows

1. **Hotel (Strong Entity):** It's identified with hotel_id. It can exist on its own.

2. **Staff (Strong Entity):** It's identified with staff_id. It can exist on its own. It can exist on its own. One hotel can consist of many staff members.

3. **Room (Strong Entity):** It's identified with room_id. It can exist on its own. One hotel can have multiple rooms.

4. **Room Types (Weak Entity):** It's partially identified with room type id because One room can only be one type. But different rooms can share the same type of Room. Without rooms, there cannot be a room type. Rooms are crucial for there to be a room type. Therefore, it's a weak entity.

5. **Breakfast Menu (Strong Entity):** It's identified with breakfast_id. It can exist on it's own since we just have create one hotel. But one hotel can have multiple breakfast menu.

6. **Breakfast Type (Weak Entity):** It's partially identified with breakfast type. One breakfast item can only be one type. But different breakfast items can be clubbed under the same category of breakfast. Without breakfast, there cannot be a breakfast type. Breakfast is crucial for there to be categorized into a type. Therefore, it's a weak entity.

7. **Coupons (Strong Entity):** It's identified with coupon_id. It can exist on its own. Even before the customer boards the room, coupons exist. Therefore, it's a strong entity. Moreover, one hotel can have multiple coupons. 1 customer can have many coupons.

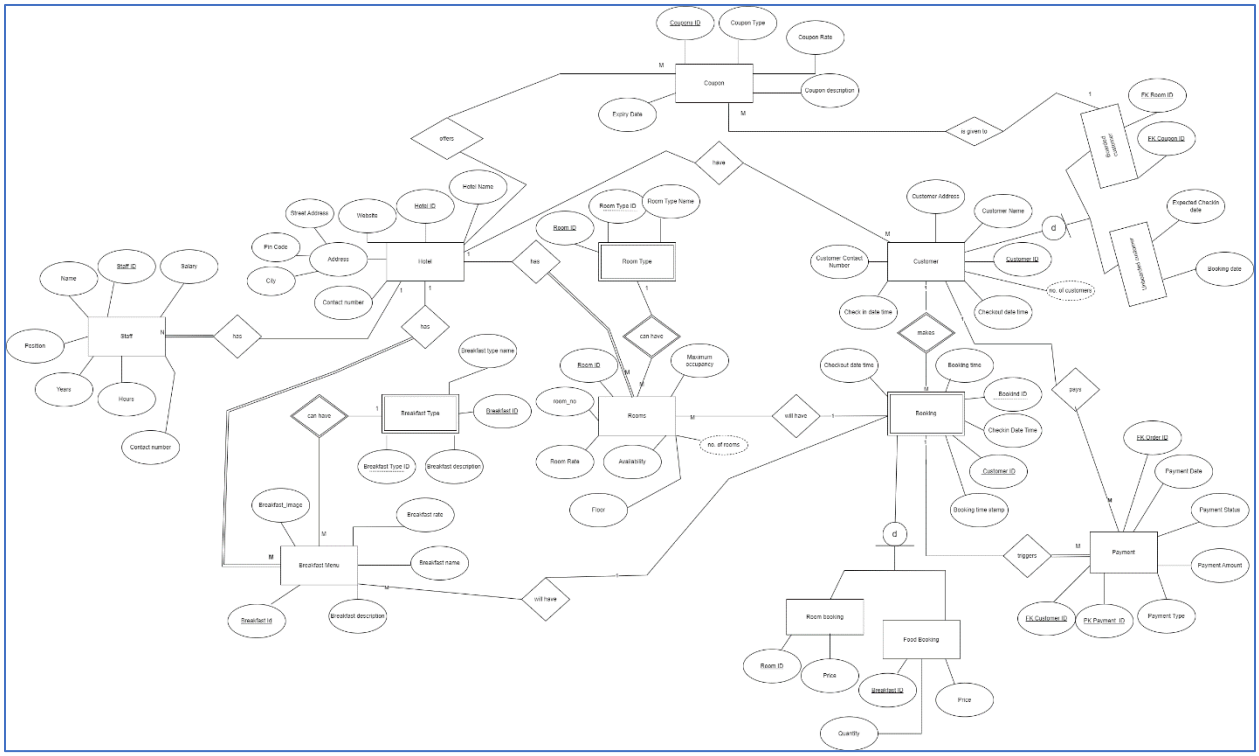
8. **Customer (Strong Entity):** It's identified with customer_id. It exists on its own. Therefore, it's a strong entity. It has two subtypes – Boarded customer and unboarded customer. Boarded customers are those customers who have already boarded and paid the amount. They will be given these coupons.

9. **Booking Order (Weak Entity):** It's partially identified by booking_id. Booking Id, requires a customer to book. There can be no order without the customer. Moreover, there can be no order without rooms and breakfast. So, it's a weak entity. It has two subtypes – Room booking and breakfast booking.

10. **Payment (Strong Entity):** It's identified by payment_id. And it has foreign key as booking_id, customer_id. Payment is associated with a booking_id.

Step 3. Now using Chen model create an ERD or EERD. Your ERD/EERD should highlight all the possible attributes for the entities. In addition, highlight what types of attributes are these, e.g., primary key, partial key, multivalued attribute etc. Note: Your ERD should contain cardinality information, and it should be built using a standard data modelling tool, such as Erwin, Visio, draw.io etc.

I have built the ERD using **draw.io.**



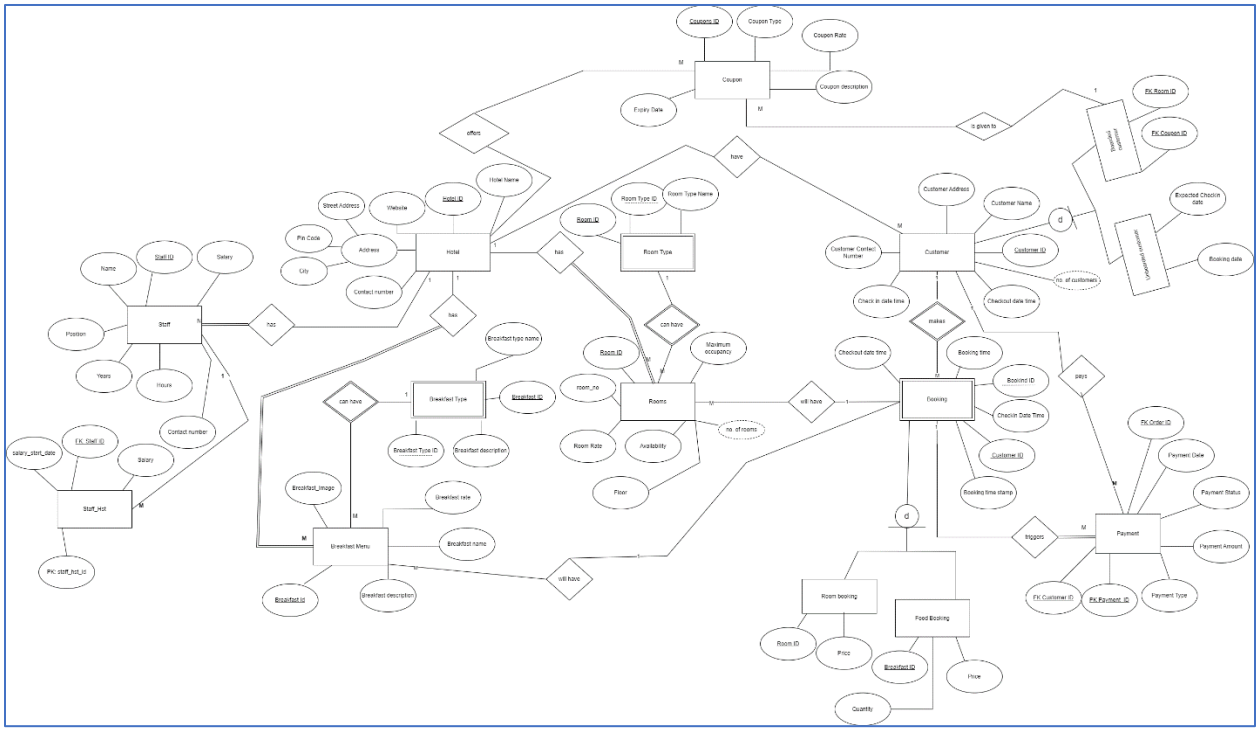
Conceptual_bedandbreakfast.io

Created using draw.io

Step 4. Identify if there are any design issues in your ERD. If you find any, then write a short paragraph about the issues, and fix it by generating a refined ERD/EERD By the end of Step 4, you will be completing a conceptual model. Now, it is time to move to Logical mapping.

Staff_Hst: staff_hst_id (Primary Key), staff_id (Foreign Key), salary_start_date, amount. There are no major issues found in the design created in step 3. But to maintain historical data of staff members' pay slip, I am adding the above entity staff_hst having attributes like described above.

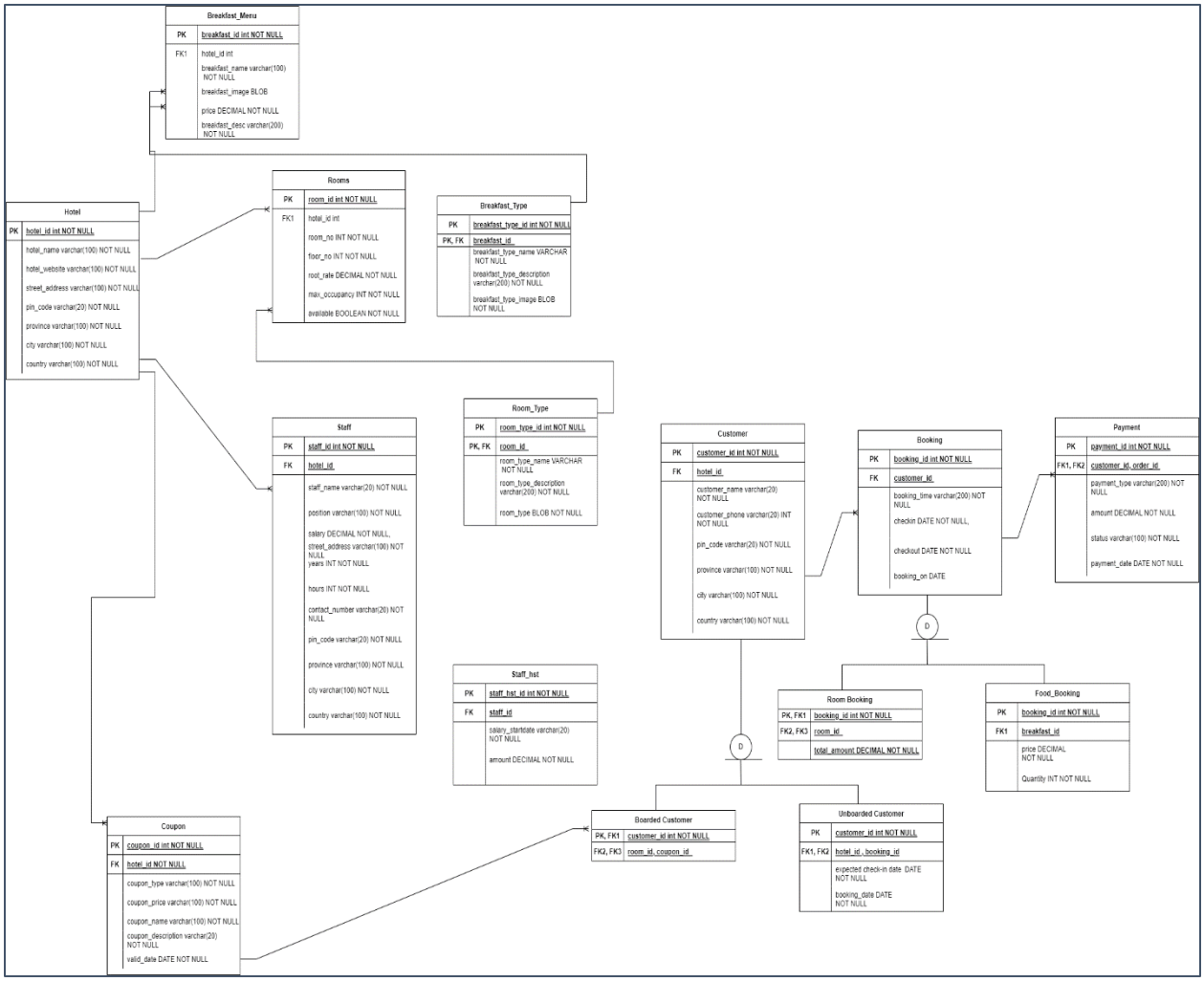
Revised conceptual ERD diagram



revised_conceptual

Created using draw.io

Step 5. Consider each entity from your data model as a table. Now, create a tabular structure with the attributes only (no need to consider any data points; just attributes are needed), and write a short paragraph and/or couple of bullet points to highlight the dependencies that exist in your logical model.



logical_model

Created using draw.io

In the **Hotel** table, hotel_id is the primary key and all the attributes are dependent on hotel_id.

In the **Rooms** Table, room_id is the primary key and all the attributes are dependent on the room_id. Here, hotel_id acts as the foreign key. One cell has one value in this table.

In the **Room_Type** table, room_id and room_type_id collectively forms the primary key and all attributes are dependent on the room_type_id. It's a weak entity.

In the **Staff** table, staff_id is the primary key and all the other attributes are dependent on the staff_id. And hotel_id acts as the foreign key. One cell has one value in this table.

In the **coupon** table, coupon_id is the primary key and all the other attributes are dependent on the coupon_id. And hotel_id acts as the foreign key. One cell has one value in this table.

In the **Breakfast_Menu** Table, breakfast_id is the primary key and all the attributes are dependent on the breakfast_id. Here, hotel_id acts as the foreign key. One cell has one value in this table.

In the **Breakfast_Type** table, breakfast_type_id and breakfast_id collectively forms the primary key and all attributes are dependent on the breakfast_type_id. It's a weak entity, dependent on the **Breakfast_Menu** table.

In the **Customer** Table, customer_id is the primary key and all other attributes are dependent on the customer _id. Here, hotel_id acts as the foreign key. One cell has one value in this table.

This has 2 subtypes:

1. Boarded customer: Here, customer_id is both the primary key and the foreign key. It has two foreign keys – room_id and couple_id
2. Unboarded customer – Here, customer_id is both the primary key and the foreign key. It has two foreign keys – hotel_id and booking_id.

In the **Booking** Table, booking_id is the primary key and all other attributes are dependent on the booking_id. Here, customer_id acts as the foreign key. One cell has one value in this table.

This has 2 subtypes

1. Room Booking – It has booking_id as the primary key and the foreign key. It also has one additional foreign key room_id. Total amount comes from the room_id.
2. Food Booking – It has booking_id as the primary key and the foreign key. It also has one additional foreign key breakfast_id. Price and Quantity comes from the breakfast_id.

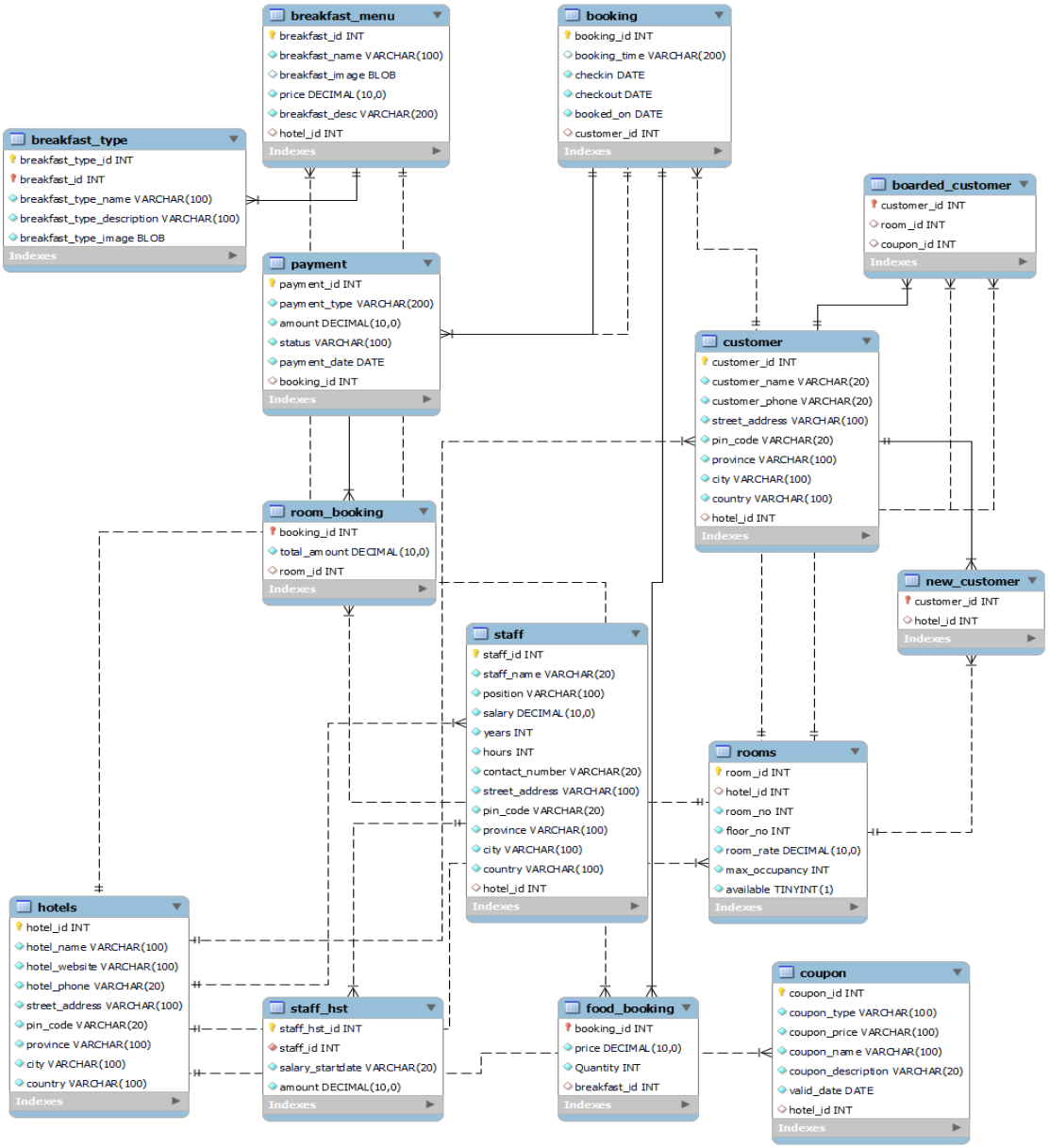
In the **Payment** table, payment_id is the primary key and customer_id and order_id acts as the foreign key. All attributes are dependent on the primary_key.

Step 6. If required, perform some level of normalization (1NF->2NF->3NF), and write your assumptions on why you considered this normalization.

It's not required for the current design.

Step 7. Now, create a database named "BedBreakfast", and create empty tables based on your final logical model by writing DDL SQL query in MySQL Workbench. This will give you the physical model.

Physical Model is as follows



Physical_model_bed_breakfast

Step 8. Export the SQL Dump of your MySQL Database and submit it as part of your assignment.

DDL SQL Query is as follows

```
CREATE SCHEMA BedBreakfast;
USE BedBreakfast;

CREATE TABLE Hotels (
   hotel_id INT PRIMARY KEY NOT NULL,
   hotel_name VARCHAR(100) NOT NULL,
   hotel_website varchar(100) NOT NULl,
   hotel_phone varchar(20) NOT NULL,
   street_address VARCHAR(100) NOT NULL,
   pin_code VARCHAR(20) NOT NULL,
   province VARCHAR(100) NOT NULL,
   city varchar(100) NOT NULL,
   country VARCHAR(100) NOT NULL
);

CREATE TABLE Rooms (
   room_id INT PRIMARY KEY,
   hotel_id INT,
   room_no INT NOT NULL,
   floor_no INT NOT NULL,
   room_rate DECIMAL NOT NULL,
   max_occupancy INT NOT NULL,
   available BOOLEAN NOT NULL,
   FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id) ON DELETE CASCADE
);
```

```
select * from hotels;

CREATE TABLE Staff(
staff_id int NOT NULL PRIMARY KEY,
staff_name varchar(20) NOT NULL,
position varchar(100) NOT NULL,
salary DECIMAL NOT NULL,
years INT NOT NULL,
hours INT NOT NULL,
contact_number varchar(20) NOT NULL,
street_address VARCHAR(100) NOT NULL,
pin_code VARCHAR(20) NOT NULL,
province VARCHAR(100) NOT NULL,
city varchar(100) NOT NULL,
country VARCHAR(100) NOT NULL,
hotel_id INT,
FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id) ON DELETE CASCADE
);

CREATE TABLE Staff_hst (
  staff_hst_id INT NOT NULL PRIMARY KEY,
  staff_id INT NOT NULL,
  salary_startdate VARCHAR(20) NOT NULL,
  amount DECIMAL NOT NULL,
  FOREIGN KEY (staff_id) REFERENCES Staff(staff_id) ON DELETE CASCADE
);
```

```
CREATE TABLE Breakfast_Menu(
breakfast_id int NOT NULL PRIMARY KEY,
breakfast_name varchar(100) NOT NULL,
breakfast_image BLOB,
price DECIMAL NOT NULL,
breakfast_desc varchar(200) NOT NULL,
hotel_id INT,
FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id) ON DELETE CASCADE
);

CREATE TABLE Breakfast_Type (
   breakfast_type_id INT,
   breakfast_id INT,
   breakfast_type_name VARCHAR(100) NOT NULL,
   breakfast_type_description VARCHAR(100) NOT NULL,
   breakfast_type_image BLOB NOT NULL,
   PRIMARY KEY (breakfast_type_id, breakfast_id),
   FOREIGN KEY (breakfast_id) REFERENCES Breakfast_Menu(breakfast_id) ON DELETE
CASCADE
);

CREATE TABLE COUPON(
   coupon_id int NOT NULL PRIMARY KEY,
   coupon_type varchar(100) NOT NULL,
   coupon_price varchar(100) NOT NULL,
```

```
    coupon_name varchar(100) NOT NULL,
    coupon_description varchar(20) NOT NULL,
    valid_date DATE NOT NULL,
    hotel_id INT,
    FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id) ON DELETE CASCADE
);


CREATE TABLE CUSTOMER(
    customer_id int NOT NULL PRIMARY KEY,
    customer_name varchar(20)NOT NULL,
    customer_phone varchar(20) NOT NULL,
    street_address VARCHAR(100) NOT NULL,
    pin_code VARCHAR(20) NOT NULL,
    province VARCHAR(100) NOT NULL,
    city varchar(100) NOT NULL,
    country VARCHAR(100) NOT NULL,
    hotel_id INT,
    FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id) ON DELETE CASCADE
);

CREATE TABLE BOARDED_CUSTOMER(
    customer_id int NOT NULL PRIMARY KEY,
    room_id INT,
    FOREIGN KEY (room_id) REFERENCES Rooms(room_id) ON DELETE CASCADE,
    coupon_id INT,
```

```
   FOREIGN KEY (coupon_id) REFERENCES Rooms(room_id) ON DELETE CASCADE,
   FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id) ON DELETE
CASCADE
);

CREATE TABLE NEW_CUSTOMER(
   customer_id int NOT NULL PRIMARY KEY,
   hotel_id INT,
   FOREIGN KEY (hotel_id) REFERENCES Rooms(hotel_id) ON DELETE CASCADE,
   FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id) ON DELETE
CASCADE
);

CREATE TABLE BOOKING(
  booking_id int NOT NULL PRIMARY KEY,
  booking_time varchar(200),
  checkin DATE NOT NULL,
  checkout DATE NOT NULL,
  booked_on DATE NOT NULL,
  customer_id INT,
  FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id) ON DELETE
CASCADE
);

CREATE TABLE Room_Booking(
booking_id int NOT NULL PRIMARY KEY,
total_amount DECIMAL NOT NULL,
```

```
room_id INT,
FOREIGN KEY (room_id) REFERENCES Rooms(room_id) ON DELETE CASCADE,
FOREIGN KEY (booking_id) REFERENCES BOOKING(booking_id) ON DELETE CASCADE
);


CREATE TABLE Food_Booking(
booking_id int NOT NULL PRIMARY KEY,
price DECIMAL NOT NULL,
Quantity INT NOT NULL,
breakfast_id INT,
FOREIGN KEY (booking_id) REFERENCES BOOKING(booking_id) ON DELETE CASCADE,
FOREIGN KEY (breakfast_id) REFERENCES Breakfast_Menu(breakfast_id) ON DELETE
CASCADE
);

CREATE TABLE PAYMENT(
  payment_id int NOT NULL PRIMARY KEY,
  payment_type varchar(200) NOT
NULL,
  amount DECIMAL NOT NULL,
  status varchar(100) NOT NULL,
  payment_date DATE NOT NULL,
  booking_id INT,
  FOREIGN KEY (booking_id) REFERENCES BOOKING(booking_id) ON DELETE CASCADE
);
```

All the sql dumps are added in the bedandbreakfast/sql_dumps/

PROBLEM 2

Follow all the steps in the given order. For you, some steps could be optional and depend on the solution you create. Note: This task does not require any web scrapping or automated information extraction from websites

Step 1. Visit https://parks.novascotia.ca/ and sub-pages within dal.ca domain. Note: You do not have to consider other provinces or the entire country. The scope is only parks of Nova Scotia.

Step 2. Identify at least 8unique entities (sub-types are not counted as separate entities for this problem; Only the supertype will be considered as 1 entity). Write your assumption in a short paragraph explaining why you considered these entities

8 unique entities are as follows

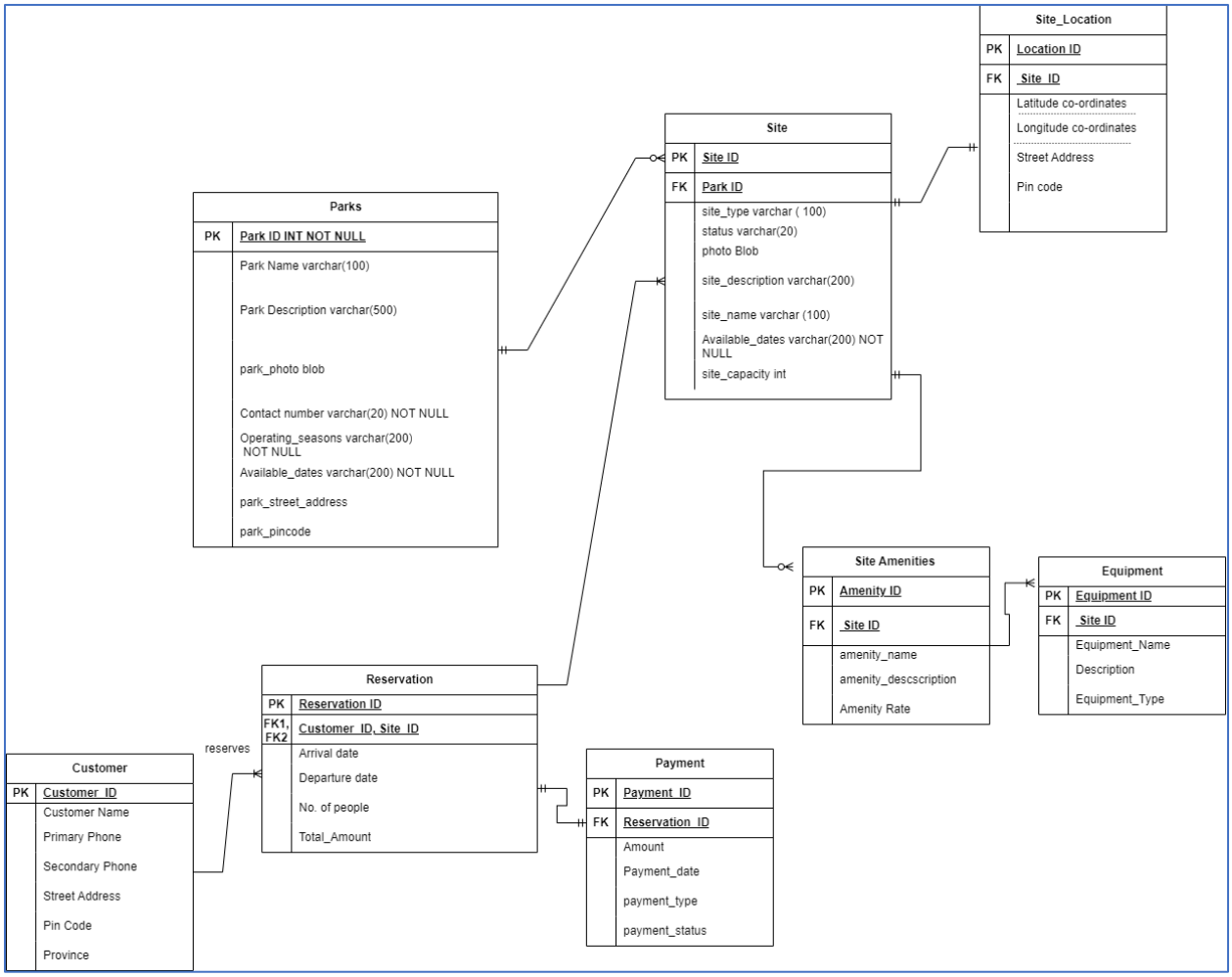
1. Parks
2. Site
3. Site Amenities
4. Equipment

5. Location
6. Customer
7. Reservation
8. Payment

I have considered these as entities basis the information I got from the website **https://parks.novascotia.ca/**. As per this website, And each site has many facilities/amenities. Sites can also be filtered using the equipment of choice. Each site is located at a particular place as shown in the list format or the map format in the website. Moreover, a customer can reserve a park and also find the available park and services of their choice. Once a customer has finalized that they want to explore this site and has reserved the spot of their choice, then they can continue with the payment option. 1 park has many sites.

That's why these entities made the reasonable choice to me. Therefore, I have chosen the above as my entities.

Step 3. Now using Crow's foot model create an ERD or EERD. Your ERD/EERD should highlight all the possible attributes for the entities. In addition, highlight what types of attributes are these, e.g., primary key, partial key, multivalued attribute etc. Note: Your ERD should contain cardinality information, and it should be built using a standard data modelling tool, such as Erwin, Visio, draw.io etc.



Conceptual_design

Created using draw.io

Step 4. Identify if there are any design issues in your ERD. If you find any, then write a short paragraph about the issues, and fix it by generating a refined ERD/EERD By the end of Step 4, you will be completing a conceptual model. Now, it is time to move to Logical mapping.

No design issues found for now.

Step 5. Consider each entity from your data model as a table. Since you have designed a crow's foot model, which has foreign key reference, and every entity is considered as a table, therefore, your design qualifies as a logical model. You do not have to perform any operation at this stage. Exception: If you perform any normalization, then you might need to decompose the structure, and you will need to provide the justification.

Logical design is same as the conceptual design

Step 6. Now, create a database, and empty tables by writing DDL SQL query in MySQL Workbench. This will give you the physical model.

DDL SQL Query is as follows:

```
CREATE SCHEMA parksnovascotia;

USE parksnovascotia;


CREATE TABLE Parks(
  park_id INT PRIMARY KEY NOT NULL,
  park_name VARCHAR(100) NOT NULL,
  park_description VARCHAR(500) NOT NULL,
  park_photo blob,
  park_contact_number VARCHAR(20) NOT NULL,
  operating_seasons VARCHAR(200) NOT NULL,
  available_dates VARCHAR(200) NOT NULL,
  park_street_address VARCHAR(200) NOT NULL,
  park_pincode VARCHAR(20) NOT NULL
);


CREATE TABLE Site(
  site_id INT PRIMARY KEY NOT NULL,
  site_type VARCHAR(100) NOT NULL,
  status VARCHAR(20) NOT NULL,
  photo BLOB,
  site_description VARCHAR(200) NOT NULL,
  site_name VARCHAR(100) NOT NULL,
```

```
  available_dates VARCHAR(200) NOT NULL,
  site_capacity INT,
  park_id INT,
  FOREIGN KEY (park_id) REFERENCES Parks(park_id) ON DELETE CASCADE
);


CREATE TABLE Site_Location(
  location_id INT PRIMARY KEY NOT NULL,
  site_street_address VARCHAR(200) NOT NULL,
  site_pincode VARCHAR(20) NOT NULL,
  site_id INT,
  FOREIGN KEY (site_id) REFERENCES Site(site_id) ON DELETE CASCADE
);

CREATE TABLE Site_Amenities(
 amenity_id INT PRIMARY KEY NOT NULL,
 amenity_name VARCHAR(50) NOT NULL,
 amenity_description VARCHAR(200) NOT NULL,
 amenity_price DECIMAL NOT NULL,
 site_id INT,
 FOREIGN KEY (site_id) REFERENCES Site(site_id) ON DELETE CASCADE
);

CREATE TABLE Equipment(
equipment_id INT PRIMARY KEY NOT NULL,
```

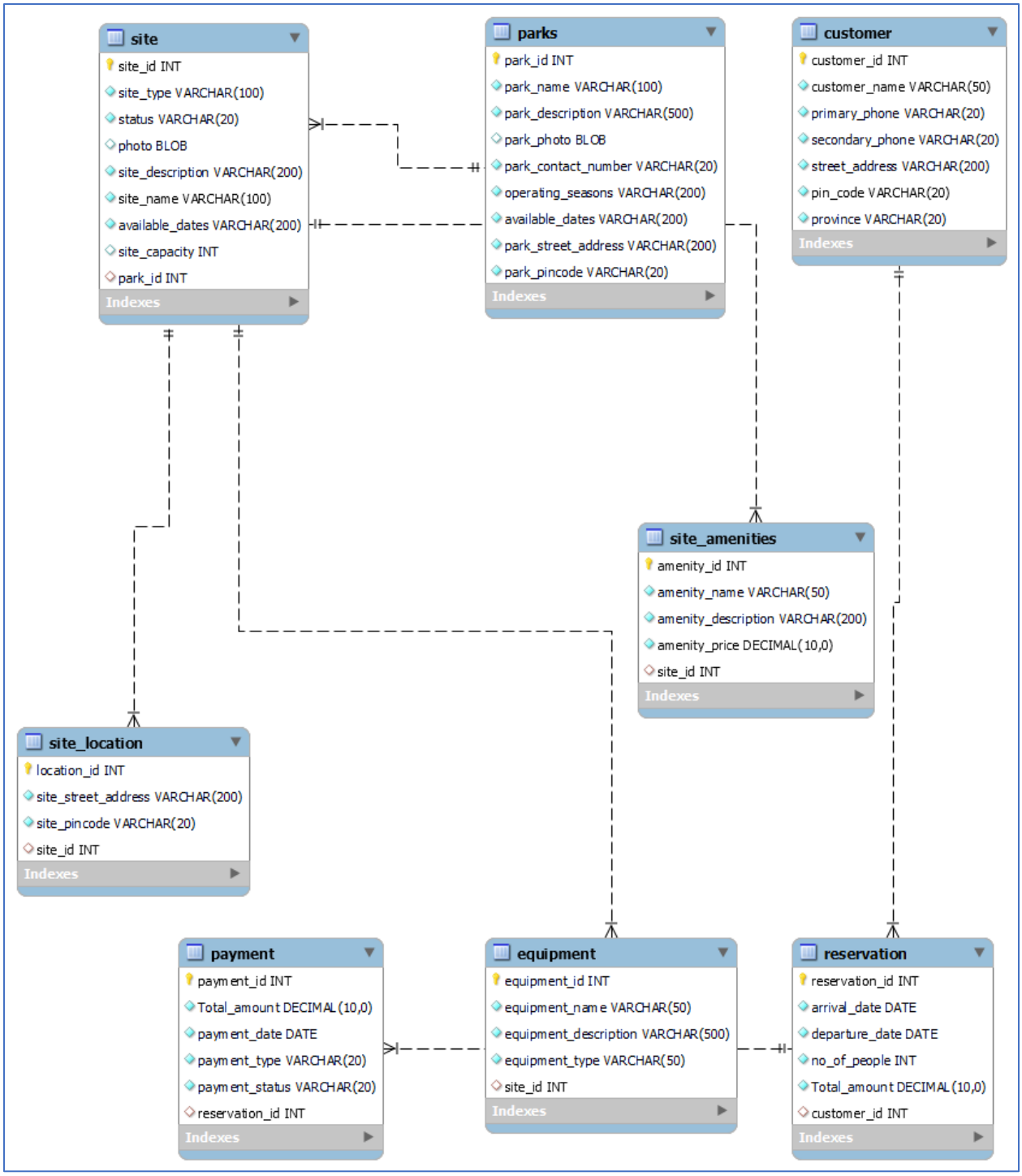
```
equipment_name VARCHAR(50) NOT NULL,
equipment_description VARCHAR(500) NOT NULL,
equipment_type VARCHAR(50) NOT NULL,
site_id INT,
FOREIGN KEY (site_id) REFERENCES Site(site_id) ON DELETE CASCADE
);

CREATE TABLE Customer(
customer_id INT PRIMARY KEY NOT NULL,
customer_name VARCHAR(50) NOT NULL,
primary_phone VARCHAR(20) NOT NULL,
secondary_phone VARCHAR(20) NOT NULL,
street_address VARCHAR(200) NOT NULL,
pin_code VARCHAR(20) NOT NULL,
province VARCHAR(20) NOT NULL
);

CREATE TABLE Reservation (
reservation_id INT PRIMARY KEY NOT NULL,
arrival_date DATE NOT NULL,
departure_date DATE NOT NULL,
no_of_people INT NOT NULL,
Total_amount DECIMAL NOT NULL,
customer_id INT,
FOREIGN KEY (customer_id) REFERENCES Customer(customer_id) ON DELETE CASCADE
);
```

```
CREATE TABLE Payment (
payment_id INT PRIMARY KEY NOT NULL,
Total_amount DECIMAL NOT NULL,
payment_date DATE NOT NULL,
payment_type VARCHAR(20) NOT NULL,
payment_status VARCHAR(20) NOT NULL,
reservation_id INT,
FOREIGN KEY (reservation_id) REFERENCES Reservation(reservation_id) ON DELETE
CASCADE
);
```

Physical Model is as follows



Physical_model_ns

Step 7. Once the physical model is built, export the SQL Dump of the database and table structure.

SQL Dumps are stored in the A1/ns_parks/sql_dumps/

Step 8. In MySQL Workbench create an ERD by performing reverse engineering. Write your observations on the similarities and differences you found between the ERD generated by you (manual), and ERD generated by MySQL Workbench (automatic).

All the tables, their cardinalities and their relationships are similar. Differences are in placement of the tables, their cardinalities and their relationships.

References

[1] C. Coronel, S. Morris, and P. Rob, "DATABASE S YSTEMS," *Crew.ee*. [Online]. Available: http://corpgov.crew.ee/Materjalid/Database%20Systems%20-%20Design,%20Implementation,%20and%20Management%20(9th%20Edition).pdf. [Accessed: 06-Jun-2023].

[2] *Mysql.com*. [Online]. Available: https://dev.mysql.com/doc/workbench/en/wb-data-modeling.html. [Accessed: 06-Jun-2023].

[3] Learning Corner, "Lecture 10 EER Diagram to relational schema," 26-Apr-2020. [Online]. Available: https://www.youtube.com/watch?v=zNt-tuVkrzY. [Accessed: 06-Jun-2023].