Three-Tier Database Application

Project Final Report
Hotel Management System (HMS)

San Jose State University

CS 157A - Introduction to Database Management Systems

CS157A-Team 13

Raza Ghulam, Kevin Pham, Bronsin Benyamin Pouran

Dr. Mike Wu

Fall 2019

Table of Contents

1.0 Introduction

Nowadays everyone has their own list of vacation destinations in mind, whether it be Cancun, the Bahamas, or the Caribbean. With so many people wanting to travel, hotels have been a classic way for vacationers to reside at while away from home. In a way, hotels are "home away from home." With that being said, it is crucial that hotel owners are able to manage their business in a clear, simple and easy manner. In addition, the booking process should be stress-free for customers who wish to stay at a hotel. For this project, we propose the development of a 3-tier architecture application that serves both the hotel owners and hotel clients. The application will take into consideration the management aspects associated with running a business, such as inventory and room availability while also factoring in the clients' needs to have a simple and concise way of booking a room.

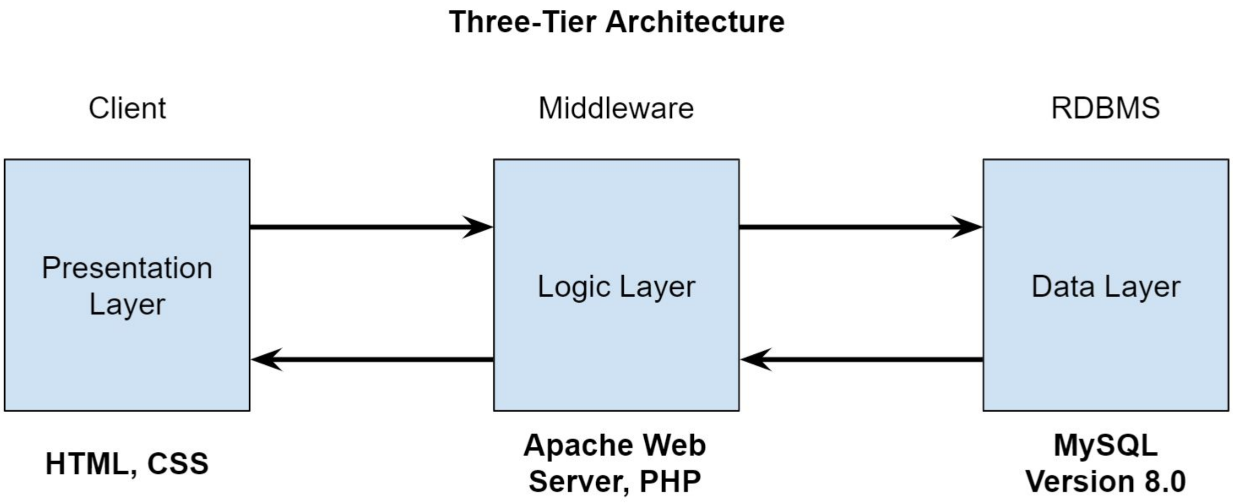
2.0 Objectives

- To provide a simple and easy to use application that provides manager and user functions to both hotel staff and clients respectively.
- Include a realistic web interface for this application
- For manager functions, be able to manage inventory and view, delete, and edit their rooms.
- For client functions, be able to view room availability, room services, and rooming fees.

3.0 Target Audiences

- Hotel Owners - Owners need a clear and simple way to manage the inventory of the hotel that they are responsible for and view, delete, and edit their rooms.
- Clients & Customers - Clients want an easy, stress-free way to book and room and have their needs catered to.

4.0 System Environment



- Presentation Layer: Users will use their preferred device and web browser to interact with the website that will be presented using HTML and CSS.
- Logic Layer: Apache Web Server will be used with server-side language, PHP.
- Data Layer: MySQL version 8.0 will be used to create, store, retrieve and manipulate tables and data from.
- Hardware/Software: Computer running either macOS or Windows will be used with IDE such as Visual Studio Code or Atom to write HTML, CSS, and PHP code. MySQL Workbench 8.0 will also be used to create and manipulate data and tables from MySQL.

5.0 Functional Requirements

The hotel management system will accomplish the following:

5.1 User authentication and authorization

The system shall provide access to the administrator, and registered members to enter the database. Administrators shall have the permission to read, write and modify the contents while member users will have only read and write permissions. The system will interact with the user as follows:

- User: Log in screen: User enters the email address or username and password.
- System: Checks the system admin or customer table and authenticate. After successful authentication, the system will authorize the user to access the contents accordingly.
- User: Enters incorrect email, username or password
- System: Prompts that the email, member or password is incorrect and if you are not a member, please sign up. The system shall allow unlimited user attempts to log in.
- User: New user shall click ‘Sign Up’ to become a member.
- System: Shall prompt the user to enter first name, last name, username, email, and password.
- User: Shall enter the above information.
- System: Will make sure that the email address is unique and does not match any existent one. After checking the uniqueness of email address, the system shall save the information in guest table and specify the user’s permissions.
- Customers: After successful login, the system will allow the user to see booking history and current booking if there is any.
- Administrators and Staff: Can create an account and log in
- Any User: Clicks log out.
- Systems: Removes his name from the current logged in users table.

5.2 Search for an available room

The database will allow both members and non members to search for available rooms. It shall also allow the authorized user to book rooms.

- User: Clicks the link ‘Home’
- System: Displays bar allowing the user to search and choose booking date and the room type.
- User: Shall search using check-in date, check-out date, room type, hotel name, and hotel address
- System: Displays the rooms based on user search.
- User: Shall select his desired room and can proceed book as long user has payment method.

5.3 Room reservation

The system will allow authorized users to reserve the available room. It shall allow the user to cancel the booking. The user shall communicate with the systems as follows:

- User: Shall click the button ‘Book’
- System: Shall display booking details and payment information
- User: After confirmation the booking detail, the user shall book by clicking ‘Book’.
- System: For non members- Shall ask them to log in or sign up.
- User: Shall provide the above information and proceed to payment.
- System: Will ensure the correctness of user information. After confirming the validity of user input, the system shall save the clients information into the guest or member table.
- Alternative: User: Shall choose cancel current booking option.
- Systems shall allow users to cancel the booking. After cancellation The systems shall update both the booked room and available room tables.

5.4 Add payment

Allow the user to put the client’s credit or debit card information. The application will validate input data and saves the information into the database.

- Users will have the option to add and save their credit card information on file for ease of access and payment.
- User: Select and click on “Add Payment”
- System: Prompt open a form will specified fields to be completed by the user. Such fields include “Name on Card”, “Card Number”, “Expiration Date”, “Security Code”.
- User: Enters and fills required fields with correct and accurate information
- User: Click on “Save”
- System: Verify that all fields have been entered in correctly
- System: Secure the information away in the database for future retrieval
- Alternative: User failed to complete all required fields, the system will prompt that all fields must be filled out

5.5 Manage rooms

The system will keep track of reserved and booked rooms. Meanwhile, the database will allow the admin to add, delete and update rooms in the system.

Note that this option is meant for management or staff use and not client use

- System: Display admins rooms
- Admin: Click on “Edit”
- System: Will display popup for admin to edit room’s status and price.
- Addit: Option to mark status as true”
- System: Will mark room as unavailable so customers don't book it
- Admin: Option to edit price
- System: Update price of room
- Admin: Click “Delete”
- System: Will delete the room.

5.6 Record and maintain inventory

The database will allow the user to add, delete and update the hotel inventory table. It will also keep a record of total hotel supplies and goods.

- System: Consists of tables for supplies, product, and miscellaneous items
- Admin: Click on link named “Inventory”
- System: Display the respective chosen option. The format will have item name listed on the left-hand side, followed by the product description, followed by the quantity on hand, followed by the number field and an update button
- Admin: Option to increase/decrease and adjust the item inventory by entering the amount in the number field and pressing the update button
- System: Take the input by the user and store the number into the database for future retrieval

5.7 Provide Price and Update

The system will allow the clients to look at various rooms and their pricing. The system will also allow managers to update price information.

- User: Members and Non-members will be able to view the prices of each rooms available in the system. The price will be available when searching for rooms, viewing a room, and during checkout.
- Administrator: Will be able to update the price of rooms when they change.

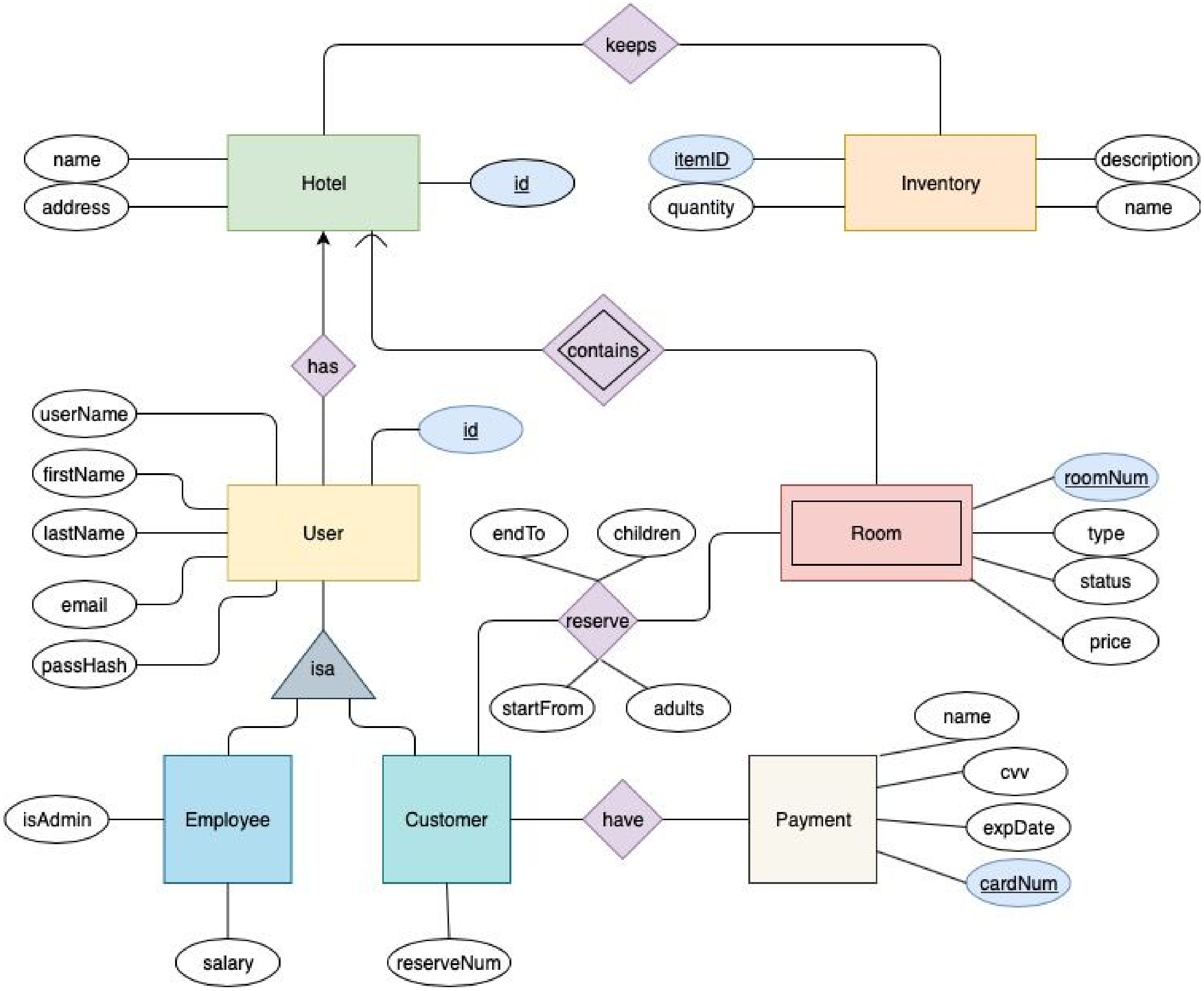
6.0 Nonfunctional Issues

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. For our HMS application, we will mainly focus on secure software tools to build our GUI and ensure database secure access. To make the database secure and reliable, we will take the following steps:

- UI Software: HTML and CSS will be used to create the User Interface of the website
- Security and Access Control:
- Login credentials with hashed passwords will be stored in the database.
- Passwords will be validated to check if they follow certain requirements to make them strong.
- Personal information such as full name, email and payment methods will be stored in the Database.
- Personal information specific to employees such as address, salary, job position and phone number will be stored in the database.
- Data displayed and functionality will depend on the type of account a user has. A user who is a customer can't access employee salary data or an employee can't change his/her own salary.
- Availability: The HMS should work with all modern web browsers and devices.

7.0 E/R Diagram

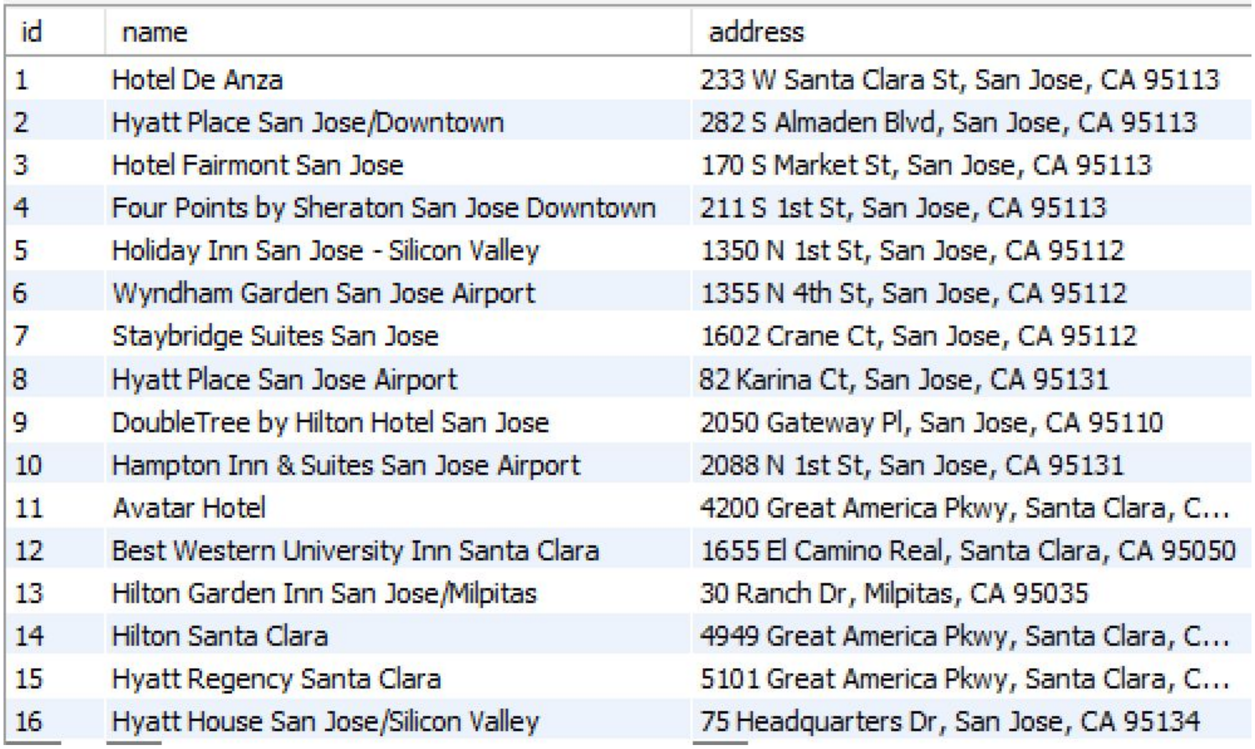
7.1 Diagram



7.2 Relational Schema

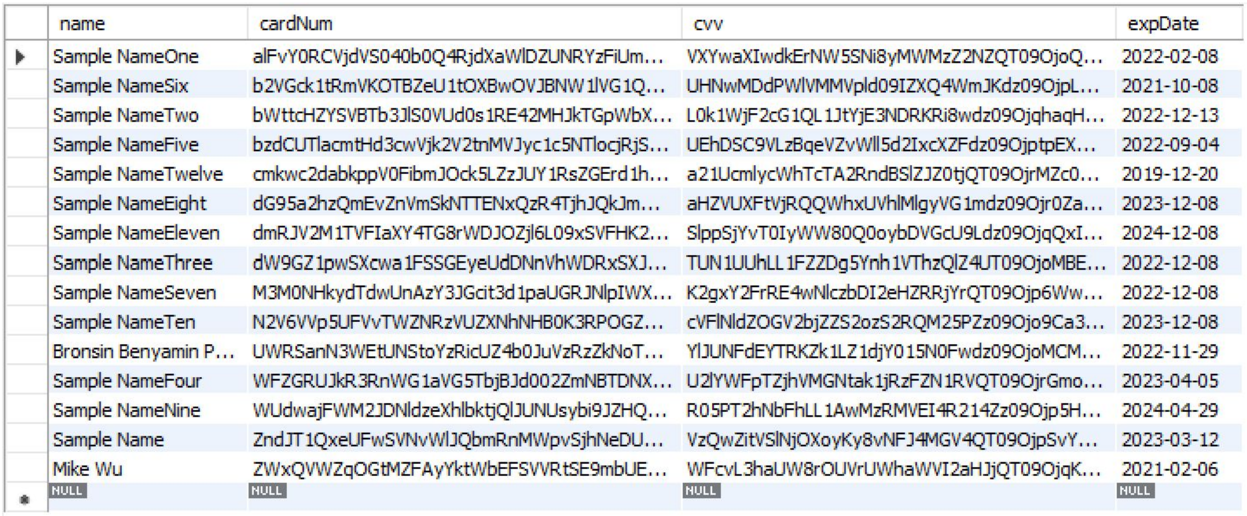
7.2.1 Entity Set

- **Hotel(id, name, address)**: Information about the hotel
    - id: Unique id of the hotel
    - name: Name of the hotel
    - address: Address of the hotel
    - **Completely Non-Trivial FDs**
        - id -> name, address
    - **Already in BCNF because left side of id -> name, address is superkey**



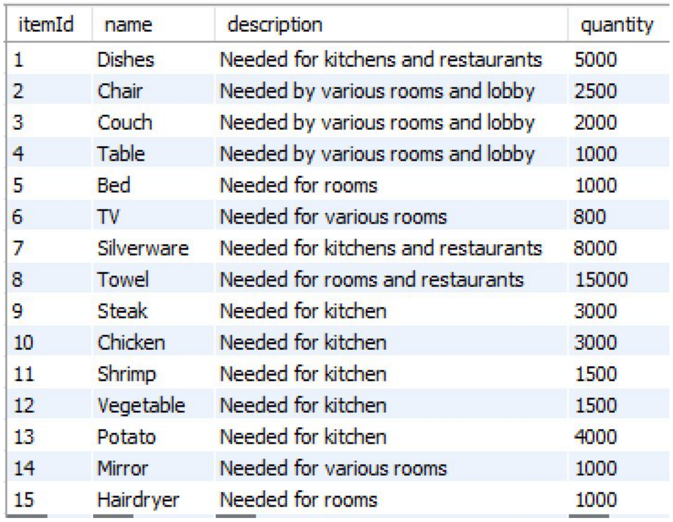
id	name	address
1	Hotel De Anza	233 W Santa Clara St, San Jose, CA 95113
2	Hyatt Place San Jose/Downtown	282 S Almaden Blvd, San Jose, CA 95113
3	Hotel Fairmont San Jose	170 S Market St, San Jose, CA 95113
4	Four Points by Sheraton San Jose Downtown	211 S 1st St, San Jose, CA 95113
5	Holiday Inn San Jose - Silicon Valley	1350 N 1st St, San Jose, CA 95112
6	Wyndham Garden San Jose Airport	1355 N 4th St, San Jose, CA 95112
7	Staybridge Suites San Jose	1602 Crane Ct, San Jose, CA 95112
8	Hyatt Place San Jose Airport	82 Karina Ct, San Jose, CA 95131
9	DoubleTree by Hilton Hotel San Jose	2050 Gateway Pl, San Jose, CA 95110
10	Hampton Inn & Suites San Jose Airport	2088 N 1st St, San Jose, CA 95131
11	Avatar Hotel	4200 Great America Pkwy, Santa Clara, C...
12	Best Western University Inn Santa Clara	1655 El Camino Real, Santa Clara, CA 95050
13	Hilton Garden Inn San Jose/Milpitas	30 Ranch Dr, Milpitas, CA 95035
14	Hilton Santa Clara	4949 Great America Pkwy, Santa Clara, C...
15	Hyatt Regency Santa Clara	5101 Great America Pkwy, Santa Clara, C...
16	Hyatt House San Jose/Silicon Valley	75 Headquarters Dr, San Jose, CA 95134

- **Payment(name, cardNum, cvv, expDate)**: Information regarding Customer's card payment
    - name: Name of the card
    - cardNum: Number of the card
    - cvv: The card's verification value
    - expDate: Expiration date of card
    - **Completely Non-Trivial FDs**
        - name -> cardNum, cvv, expDate
    - **Already in BCNF because left side of name -> cardNum, cvv, expDate is a superkey**



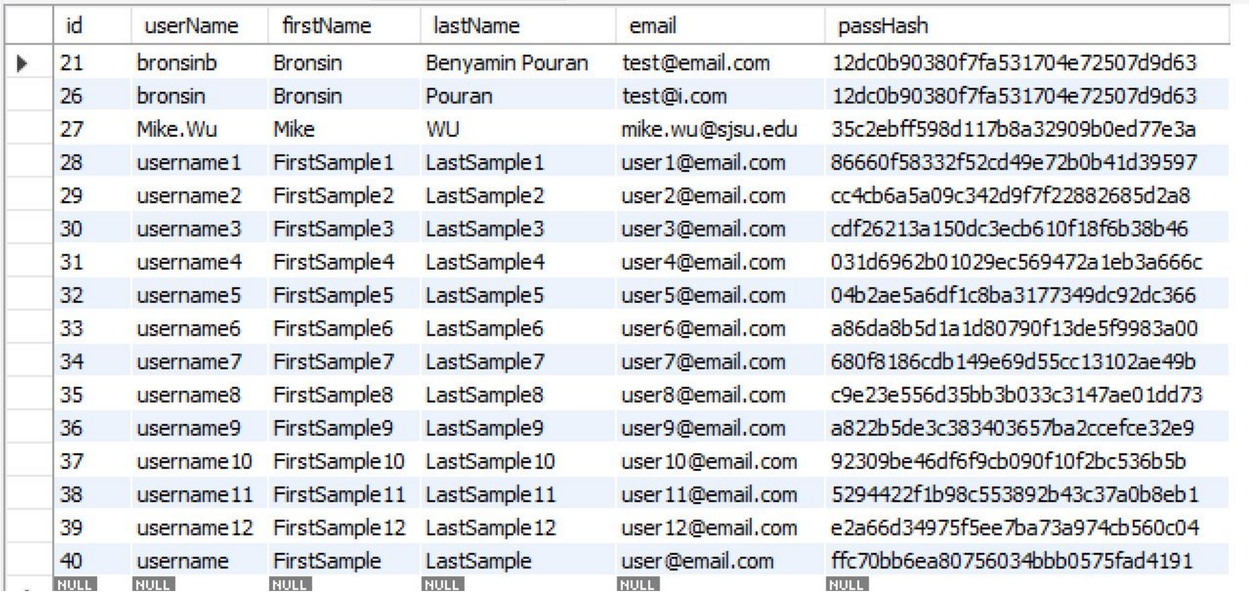
	name	cardNum	cvv	expDate
▶	Sample NameOne	alFvY0RCVjdVS040b0Q4RjdXaWlDZUNRYzFiUm…	VXYwaXIwdkErNW5SNi8yMWMzZ2NZQT09OjoQ…	2022-02-08
	Sample NameSix	b2VGck1tRmVKOTBZeU1tOXBwOVJBNW1lVG1Q…	UHNwMDdPWlVMMVpld09IZXQ4WmJKdz09OjpL…	2021-10-08
	Sample NameTwo	bWttcHZYSVBTb3JlS0VUd0s1RE42MHJkTGpWbX…	L0k1WjF2cG1QL1JtYjE3NDRKRi8wdz09OjqhaqH…	2022-12-13
	Sample NameFive	bzdCUTlacmtHd3cwVjk2V2tnMVJyc1c5NTlocjRjS…	UEhDSC9VLzBqeVZvWll5d2IxcXZFdz09OjptpEX…	2022-09-04
	Sample NameTwelve	cmkwc2dabkppV0FibmJOck5LZzJUY1RsZGErd1h…	a21UcmlycWhTcTA2RndBSlZJZ0tjQT09OjrMZc0…	2019-12-20
	Sample NameEight	dG95a2hzQmEvZnVmSkNTTENxQzR4TjhJQkJm…	aHZVUXFtVjRQQWhxUVhlMlgyVG1mdz09Ojr0Za…	2023-12-08
	Sample NameEleven	dmRJV2M1TVFIaXY4TG8rWDJOZjl6L09xSVFHK2…	SlppSjYvT0IyWW80Q0oybDVGcU9Ldz09OjqQxI…	2024-12-08
	Sample NameThree	dW9GZ1pwSXcwa1FSSGEyeUdDNnVhWDRxSXJ…	TUN1UUhLL1FZZDg5Ynh1VThzQlZ4UT09OjoMBE…	2022-12-08
	Sample NameSeven	M3M0NHkydTdwUnAzY3JGcit3d1paUGRJNlpIWX…	K2gxY2FrRE4wNlczbDI2eHZRRjYrQT09Ojp6Ww…	2022-12-08
	Sample NameTen	N2V6VVp5UFVvTWZNRzVUZXNhNHB0K3RPOGZ…	cVFlNldZOGV2bjZZS2ozS2RQM25PZz09Ojo9Ca3…	2023-12-08
	Bronsin Benyamin P…	UWRSanN3WEtUNStoYzRicUZ4b0JuVzRzZkNoT…	YlJUNFdEYTRKZk1LZ1djY015N0Fwdz09OjoMCM…	2022-11-29
	Sample NameFour	WFZGRUJkR3RnWG1aVG5TbjBJd002ZmNBTDNX…	U2lYWFpTZjhVMGNtak1jRzFZN1RVQT09OjrGmo…	2023-04-05
	Sample NameNine	WUdwajFWM2JDNldzeXhlbktjQlJUNUsybi9JZHQ…	R05PT2hNbFhLL1AwMzRMVEI4R214Zz09Ojp5H…	2024-04-29
	Sample Name	ZndJT1QxeUFwSVNvWlJQbmRnMWpvSjhNeDU…	VzQwZitVSlNjOXoyKy8vNFJ4MGV4QT09OjpSvY…	2023-03-12
	Mike Wu	ZWxQVWZqOGtMZFAyYktWbEFSVVRtSE9mbUE…	WFcvL3haUW8rOUVrUWhaWVI2aHJjQT09OjqK…	2021-02-06
✱	NULL	NULL	NULL	NULL

- **Inventory(<u>itemID</u>, name, Description, quantity)**: Information about the items used for upkeeping the hotel
  - <u>itemID</u>: Unique id of the inventory
  - Name: Name of the item
  - Description: Describes what the inventory is
  - Quantity: The amount of left of the inventory
  - **Completely Non-Trivial FDs**
    - itemId -> name, description, quantity
  - **Already in BCNF because left side of itemId -> name, description, quantity is a superkey**



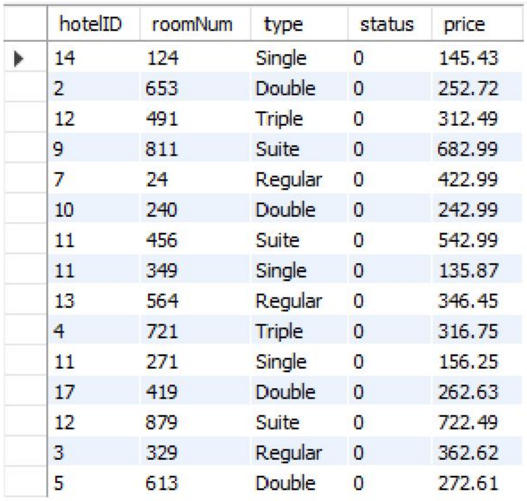
itemId	name	description	quantity
1	Dishes	Needed for kitchens and restaurants	5000
2	Chair	Needed by various rooms and lobby	2500
3	Couch	Needed by various rooms and lobby	2000
4	Table	Needed by various rooms and lobby	1000
5	Bed	Needed for rooms	1000
6	TV	Needed for various rooms	800
7	Silverware	Needed for kitchens and restaurants	8000
8	Towel	Needed for rooms and restaurants	15000
9	Steak	Needed for kitchen	3000
10	Chicken	Needed for kitchen	3000
11	Shrimp	Needed for kitchen	1500
12	Vegetable	Needed for kitchen	1500
13	Potato	Needed for kitchen	4000
14	Mirror	Needed for various rooms	1000
15	Hairdryer	Needed for rooms	1000

- **User(<u>id,</u> userName, firstName, lastName, email, passHash)**: Login information of database users
  - <u>id</u>: Unique ID of the database user
  - userName: The username of the user
  - firstName: First name of the user
  - lastName: Last name of the user
  - email: Email of the user
  - passHash: Hashed password of user for authentication
  - **Completely Non-Trivial FDs**
    - id -> username, firstName, lastName, email, passHash
  - **Already in BCNF because left side of id -> username, firstName, lastName, email, passHash is a superkey**



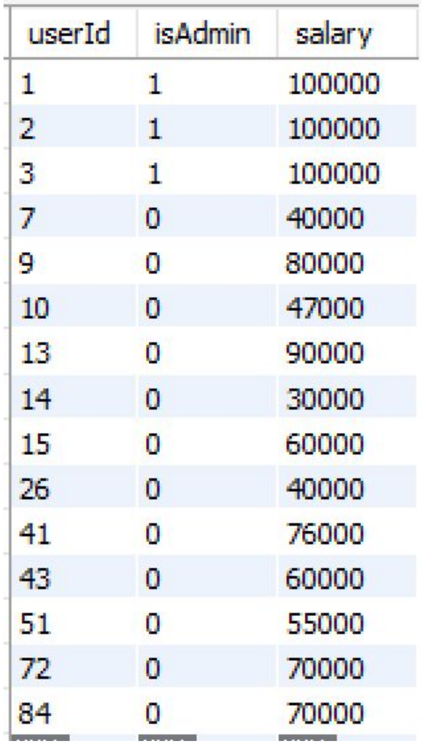
	id	userName	firstName	lastName	email	passHash
▶	21	bronsinb	Bronsin	Benyamin Pouran	test@email.com	12dc0b90380f7fa531704e72507d9d63
	26	bronsin	Bronsin	Pouran	test@i.com	12dc0b90380f7fa531704e72507d9d63
	27	Mike.Wu	Mike	WU	mike.wu@sjsu.edu	35c2ebff598d117b8a32909b0ed77e3a
	28	username1	FirstSample1	LastSample1	user1@email.com	86660f58332f52cd49e72b0b41d39597
	29	username2	FirstSample2	LastSample2	user2@email.com	cc4cb6a5a09c342d9f7f22882685d2a8
	30	username3	FirstSample3	LastSample3	user3@email.com	cdf26213a150dc3ecb610f18f6b38b46
	31	username4	FirstSample4	LastSample4	user4@email.com	031d6962b01029ec569472a1eb3a666c
	32	username5	FirstSample5	LastSample5	user5@email.com	04b2ae5a6df1c8ba3177349dc92dc366
	33	username6	FirstSample6	LastSample6	user6@email.com	a86da8b5d1a1d80790f13de5f9983a00
	34	username7	FirstSample7	LastSample7	user7@email.com	680f8186cdb149e69d55cc13102ae49b
	35	username8	FirstSample8	LastSample8	user8@email.com	c9e23e556d35bb3b033c3147ae01dd73
	36	username9	FirstSample9	LastSample9	user9@email.com	a822b5de3c383403657ba2ccefce32e9
	37	username10	FirstSample10	LastSample10	user10@email.com	92309be46df6f9cb090f10f2bc536b5b
	38	username11	FirstSample11	LastSample11	user11@email.com	5294422f1b98c553892b43c37a0b8eb1
	39	username12	FirstSample12	LastSample12	user12@email.com	e2a66d34975f5ee7ba73a974cb560c04
	40	username	FirstSample	LastSample	user@email.com	ffc70bb6ea80756034bbb0575fad4191
	NULL	NULL	NULL	NULL	NULL	NULL

- **Room(<u>HotelId</u>, <u>roomNum</u>, type, status, price)**: Information about room
  - <u>HotelId</u>: The primary key of the hotel
  - <u>roomNum</u>: Unique room number of the hotel
  - type: The type of room (Single, Double, Triple, Regular, Suite)
  - status: Availability of the room (Booked, Available)
  - Price: The price of the room
  - **Completely Non-Trivial FDs**
    - HotelId, roomNum -> type, status, price
  - **Already in BCNF because left side of HotelId, roomNum -> type, status, price is a superkey**



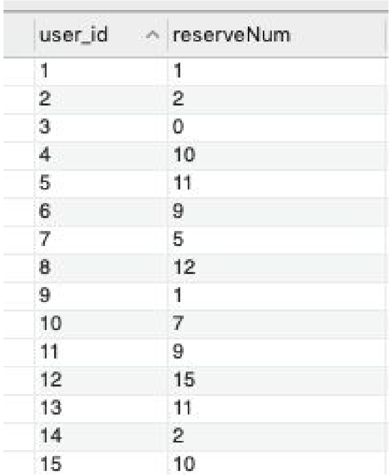
hotelID	roomNum	type	status	price
14	124	Single	0	145.43
2	653	Double	0	252.72
12	491	Triple	0	312.49
9	811	Suite	0	682.99
7	24	Regular	0	422.99
10	240	Double	0	242.99
11	456	Suite	0	542.99
11	349	Single	0	135.87
13	564	Regular	0	346.45
4	721	Triple	0	316.75
11	271	Single	0	156.25
17	419	Double	0	262.63
12	879	Suite	0	722.49
3	329	Regular	0	362.62
5	613	Double	0	272.61

- **Employee(<u>userId</u>, isAdmin, salary)**: Hotel employee user with information about his employment
    - <u>userId</u>: ID from superclass user
    - isAdmin: Defines if the employee is an admin (privileges)
    - salary: Employee’s monthly salary
    - **Completely Non-Trivial FDs**
        - userId -> isAdmin, salary
    - **Already in BCNF because left side of userId -> isAdmin, salary is a superkey**



userId	isAdmin	salary
1	1	100000
2	1	100000
3	1	100000
7	0	40000
9	0	80000
10	0	47000
13	0	90000
14	0	30000
15	0	60000
26	0	40000
41	0	76000
43	0	60000
51	0	55000
72	0	70000
84	0	70000

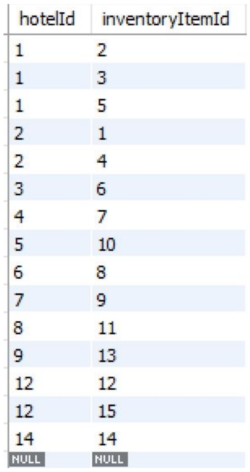
- **Customer(user_Id, reserveNum)**: Keeps track of returning customers
    - user_Id: ID from superclass user
    - reserveNum: Amount of times customer booked/reserved a room with hotel
    - **Completely Non-Trivial FDs**
        - user_id -> reserveNum
    - **Already in BCNF because left side of user_id -> reserveNum is a superkey**



user_id	reserveNum
1	1
2	2
3	0
4	10
5	11
6	9
7	5
8	12
9	1
10	7
11	9
12	15
13	11
14	2
15	10

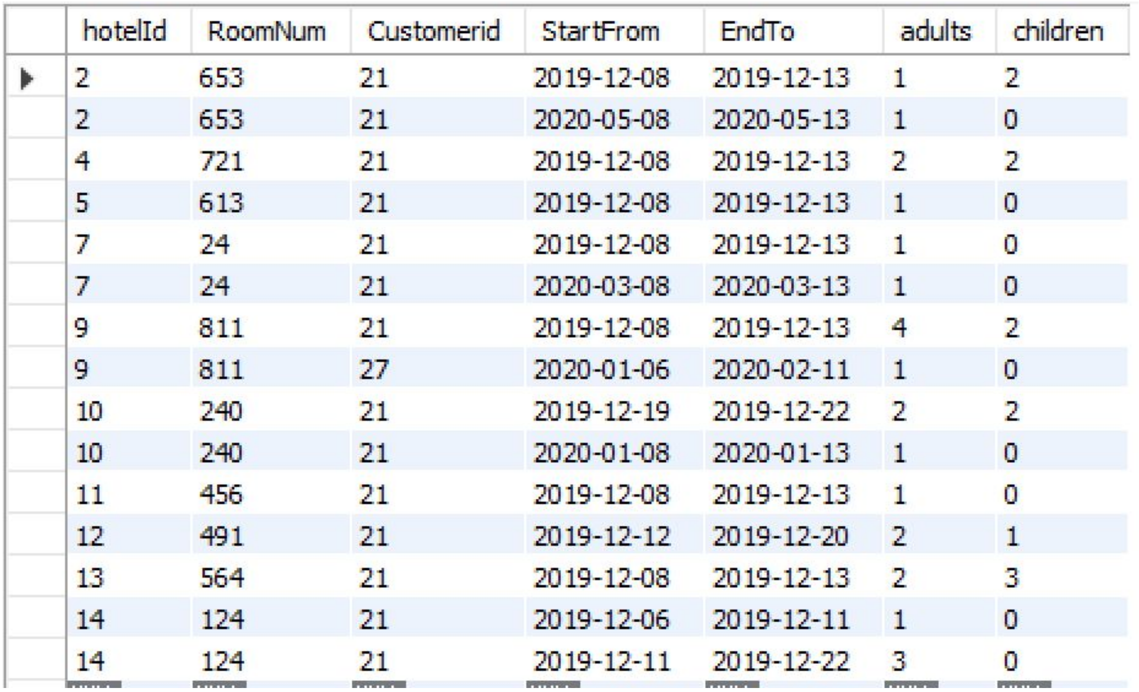
7.2.2 Relationships

- **keep(HotelId, InventoryItemID):** Relationship between hotel and its inventory
    - HotelId: The primary key of the hotel
    - InventoryItemID: The primary key of the Inventory
    - **Completely Non-Trivial FDs**
        - inventoryItemId -> hotelId
    - **Not in BCNF because left side of inventory inventoryItemId -> hotelId is not superkey, however, no decomposition needed because left side is prime making it 3NF**



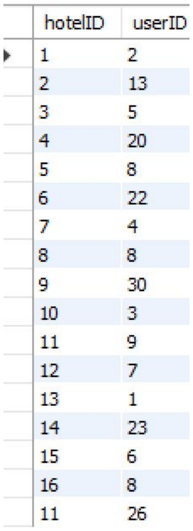
hotelId	inventoryItemId
1	2
1	3
1	5
2	1
2	4
3	6
4	7
5	10
6	8
7	9
8	11
9	13
12	12
12	15
14	14
NULL	NULL

- **reserve(Customerid, hotelId, RoomNum, StartFrom, EndTo, adults, childern)**: Information in relation to customer, its room and the time period the room will be booked for.
    - CustomerId: the guest reserve number
    - hotelId: Id of hotel that room belongs to
    - RoomNum: The guest's room number
    - StartFrom: Check in date
    - EndTo: Check out date
    - adults: Number of adults
    - children: number of adults
    - **Completely Non-Trivial FDs**
        - hotelId, RoomNum, StartFrom, EndTo, CustomerId -> adults, children
    - **Already in BCNF because left side of inventory hotelId, RoomNum, StartFrom, EndTo, CustomerId -> adults, children is a superkey**



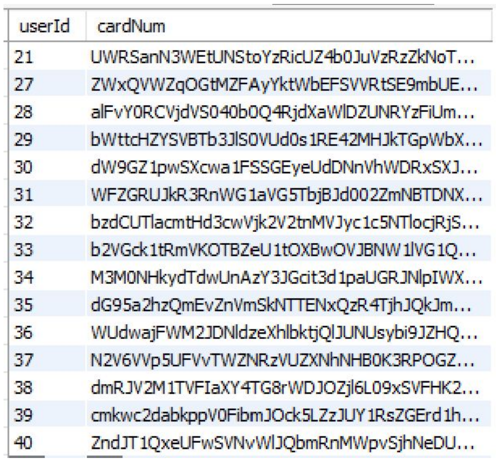
hotelId	RoomNum	Customerid	StartFrom	EndTo	adults	children
2	653	21	2019-12-08	2019-12-13	1	2
2	653	21	2020-05-08	2020-05-13	1	0
4	721	21	2019-12-08	2019-12-13	2	2
5	613	21	2019-12-08	2019-12-13	1	0
7	24	21	2019-12-08	2019-12-13	1	0
7	24	21	2020-03-08	2020-03-13	1	0
9	811	21	2019-12-08	2019-12-13	4	2
9	811	27	2020-01-06	2020-02-11	1	0
10	240	21	2019-12-19	2019-12-22	2	2
10	240	21	2020-01-08	2020-01-13	1	0
11	456	21	2019-12-08	2019-12-13	1	0
12	491	21	2019-12-12	2019-12-20	2	1
13	564	21	2019-12-08	2019-12-13	2	3
14	124	21	2019-12-06	2019-12-11	1	0
14	124	21	2019-12-11	2019-12-22	3	0

- **has(<u>HotelId</u>, <u>UserId</u>)**: Relationship between hotel and its users
  - <u>HotelId</u>: The primary key of the hotel
  - <u>UserId</u>: The primary key of the hotel
  - **Completely Non-Trivial FDs**
    - userId -> hotelId
  - **Not in BCNF because left side of inventory userId -> hotelId is not superkey, however, no decomposition needed because left side is prime making it 3NF**



hotelID	userID
1	2
2	13
3	5
4	20
5	8
6	22
7	4
8	8
9	30
10	3
11	9
12	7
13	1
14	23
15	6
16	8
11	26

- **have(<u>userId, cardNum</u>)**: Relationship between customer and their payment information
  - <u>userId:</u> The primary key of customer
  - <u>cardNum</u>: The primary key of the payment
  - **Completely Non-Trivial FDs**
    - userId -> cardNum
  - **Not in BCNF because left side of inventory userId -> cardNum is not superkey, however, no decomposition needed because left side is prime making it 3NF**

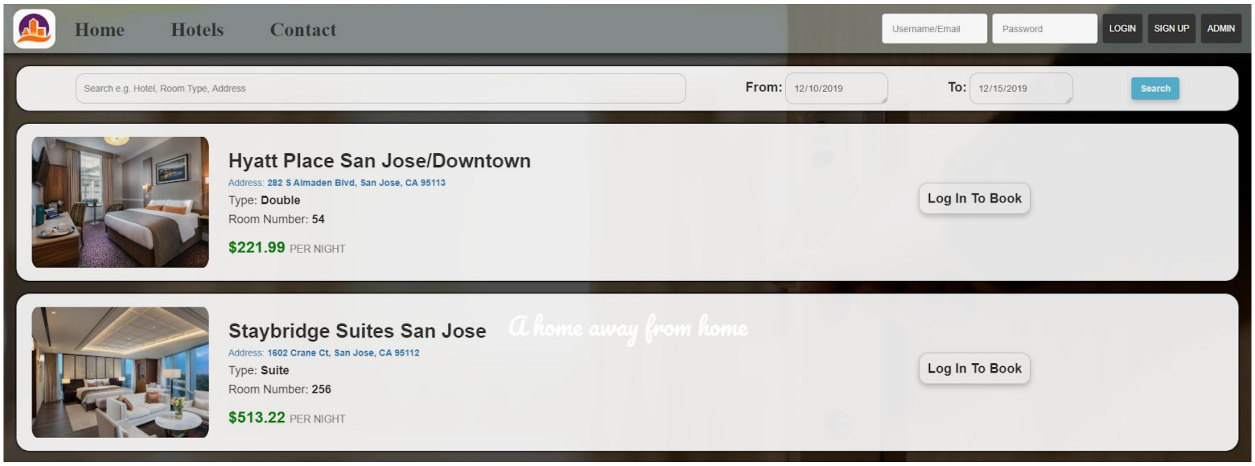


userId	cardNum
21	UWRSanN3WEtUNStoYzRicUZ4b0JuVzRzZkNoT...
27	ZWxQVWZqOGtMZFAyYktWbEFSVVRtSE9mbUE...
28	alFvY0RCVjdVS040b0Q4RjdXaWlDZUNRYzFiUm...
29	bWttcHZYSVBTb3JlS0VUd0s1RE42MHJkTGpWbX...
30	dW9GZ1pwSXcwa1FSSGEyeUdDNnVhWDRxSXJ...
31	WFZGRUJkR3RnWG1aVG5TbjBJd002ZmNBTDNX...
32	bzdCUTlacmtHd3cwVjk2V2tnMVJyc1c5NTlocjRjS...
33	b2VGck1tRmVKOTBZeU1tOXBwOVJBNW1lVG1Q...
34	M3M0NHkydTdwUnAzY3JGcit3d1paUGRJNlpIWX...
35	dG95a2hzQmEvZnVmSkNTTENxQzR4TjhJQkJm...
36	WUdwajFWM2JDNldzeXhlbktjQlJUNUsybi9JZHQ...
37	N2V6VVp5UFVvTWZNRzVUZXNhNHB0K3RPOGZ...
38	dmRJV2M1TVFIaXY4TG8rWDJOZjl6L09xSVFHK2...
39	cmkwc2dabkppV0FibmJOck5LZzJUY1RsZGErd1h...
40	ZndJT1QxeUFwSVNvWlJQbmRnMWpvSjhNeDU...

8.0 Implementation and Screen Shots

8.1 Customer Sign Up

- Landing Page

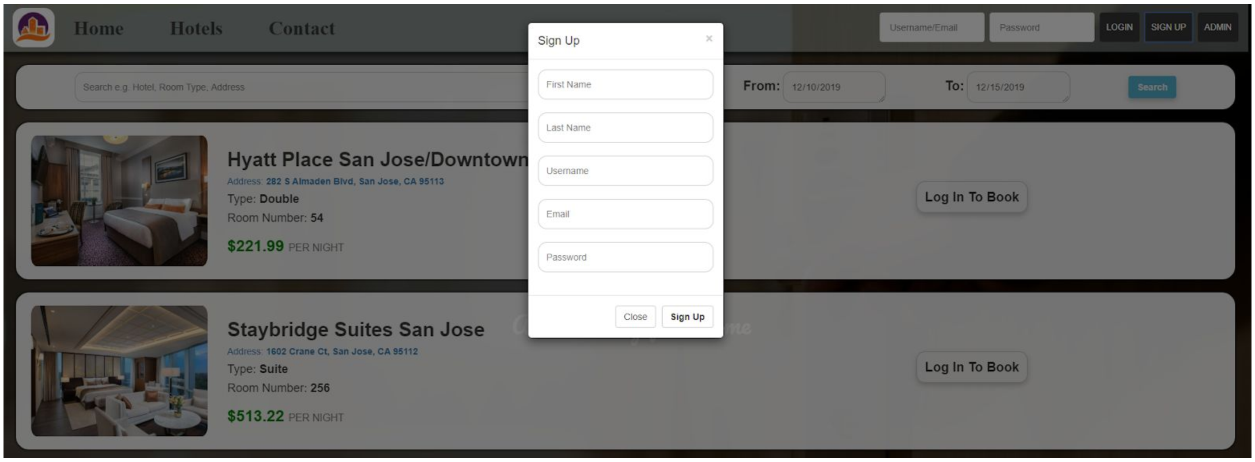


From: 12/10/2019 To: 12/15/2019

Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Type: Double
Room Number: 54
$221.99 PER NIGHT

Staybridge Suites San Jose
Address: 1602 Crane Ct, San Jose, CA 95112
Type: Suite
Room Number: 256
$513.22 PER NIGHT

- Sign Up

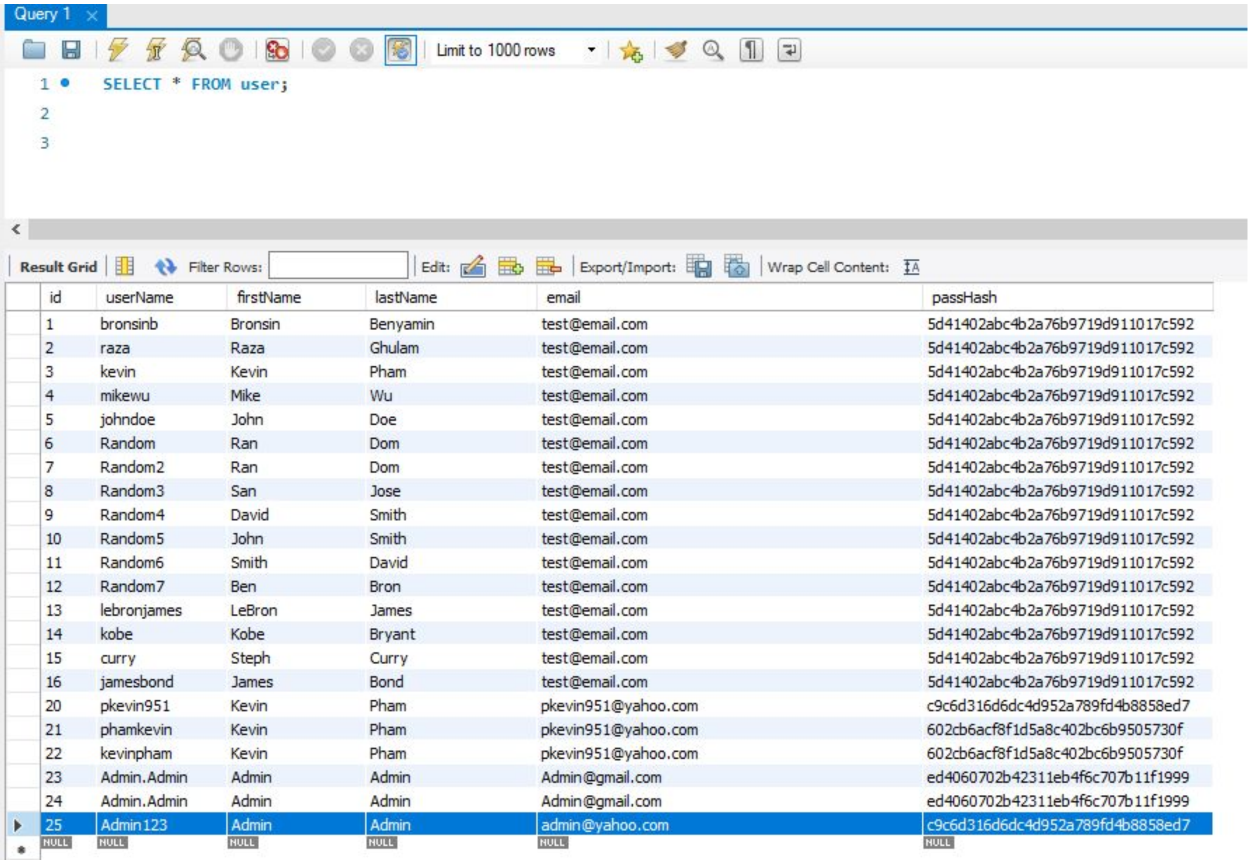


From: 12/10/2019 To: 12/15/2019

Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Type: Double
Room Number: 54
$221.99 PER NIGHT

Staybridge Suites San Jose
Address: 1602 Crane Ct, San Jose, CA 95112
Type: Suite
Room Number: 256
$513.22 PER NIGHT

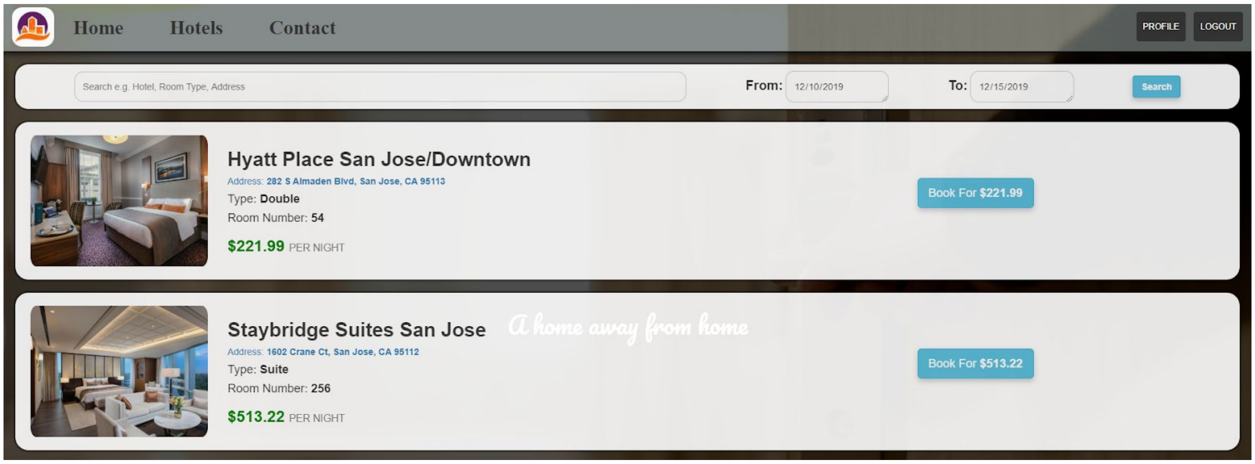
- mySQL of newly added user



```
SELECT * FROM user;
```

id	userName	firstName	lastName	email	passHash
1	bronsinb	Bronsin	Benyamin	test@email.com	5d41402abc4b2a76b9719d911017c592
2	raza	Raza	Ghulam	test@email.com	5d41402abc4b2a76b9719d911017c592
3	kevin	Kevin	Pham	test@email.com	5d41402abc4b2a76b9719d911017c592
4	mikewu	Mike	Wu	test@email.com	5d41402abc4b2a76b9719d911017c592
5	johndoe	John	Doe	test@email.com	5d41402abc4b2a76b9719d911017c592
6	Random	Ran	Dom	test@email.com	5d41402abc4b2a76b9719d911017c592
7	Random2	Ran	Dom	test@email.com	5d41402abc4b2a76b9719d911017c592
8	Random3	San	Jose	test@email.com	5d41402abc4b2a76b9719d911017c592
9	Random4	David	Smith	test@email.com	5d41402abc4b2a76b9719d911017c592
10	Random5	John	Smith	test@email.com	5d41402abc4b2a76b9719d911017c592
11	Random6	Smith	David	test@email.com	5d41402abc4b2a76b9719d911017c592
12	Random7	Ben	Bron	test@email.com	5d41402abc4b2a76b9719d911017c592
13	lebronjames	LeBron	James	test@email.com	5d41402abc4b2a76b9719d911017c592
14	kobe	Kobe	Bryant	test@email.com	5d41402abc4b2a76b9719d911017c592
15	curry	Steph	Curry	test@email.com	5d41402abc4b2a76b9719d911017c592
16	jamesbond	James	Bond	test@email.com	5d41402abc4b2a76b9719d911017c592
20	pkevin951	Kevin	Pham	pkevin951@yahoo.com	c9c6d316d6dc4d952a789fd4b8858ed7
21	phamkevin	Kevin	Pham	pkevin951@yahoo.com	602cb6acf8f1d5a8c402bc6b9505730f
22	kevinpham	Kevin	Pham	pkevin951@yahoo.com	602cb6acf8f1d5a8c402bc6b9505730f
23	Admin.Admin	Admin	Admin	Admin@gmail.com	ed4060702b42311eb4f6c707b11f1999
24	Admin.Admin	Admin	Admin	Admin@gmail.com	ed4060702b42311eb4f6c707b11f1999
25	Admin123	Admin	Admin	admin@yahoo.com	c9c6d316d6dc4d952a789fd4b8858ed7
NULL	NULL	NULL	NULL	NULL	NULL

- Landing Page once logged in



From: 12/10/2019 To: 12/15/2019

Hyatt Place San Jose/Downtown

Address: 282 S Almaden Blvd, San Jose, CA 95113

Type: Double

Room Number: 54

$221.99 PER NIGHT

Book For $221.99

Staybridge Suites San Jose

Address: 1602 Crane Ct, San Jose, CA 95112

Type: Suite

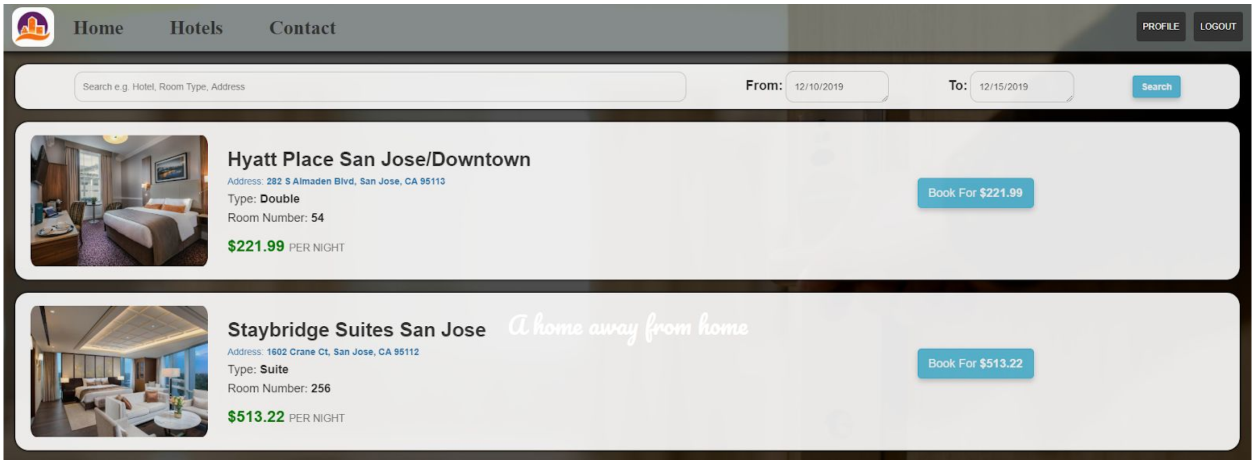
Room Number: 256

$513.22 PER NIGHT

Book For $513.22

8.2 Customer Login

- Landing page once logged in

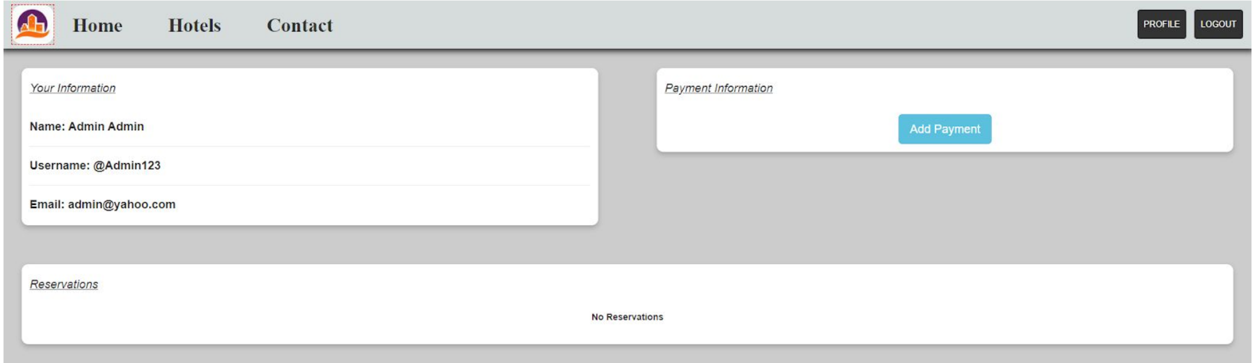


From: 12/10/2019 To: 12/15/2019

Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Type: Double
Room Number: 54
$221.99 PER NIGHT

Staybridge Suites San Jose
Address: 1602 Crane Ct, San Jose, CA 95112
Type: Suite
Room Number: 256
$513.22 PER NIGHT

- Profile



Your Information

Name: Admin Admin

Username: @Admin123

Email: admin@yahoo.com

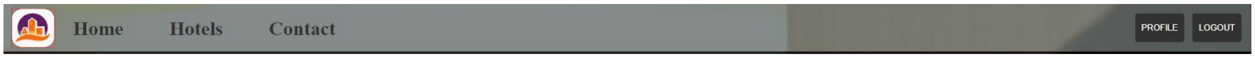
Payment Information

Reservations

No Reservations

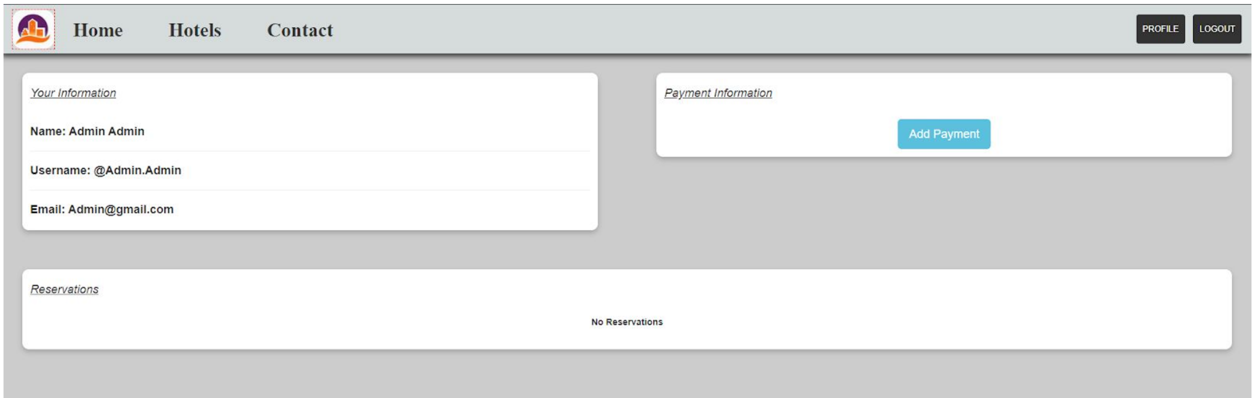
8.3 Payment

- Select Profile to add payment



8.3.1 Add Payment

- Profile Landing page



Your Information

Name: Admin Admin

Username: @Admin.Admin

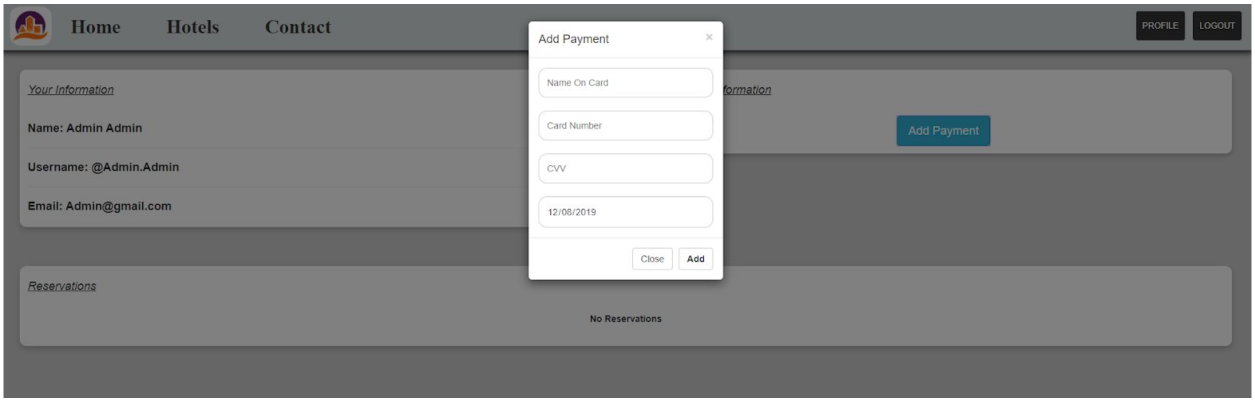
Email: Admin@gmail.com

Payment Information

Reservations

No Reservations

- Add Payment (Blue Button)



Add Payment

Name On Card

Card Number

CVV

12/08/2019

Your Information

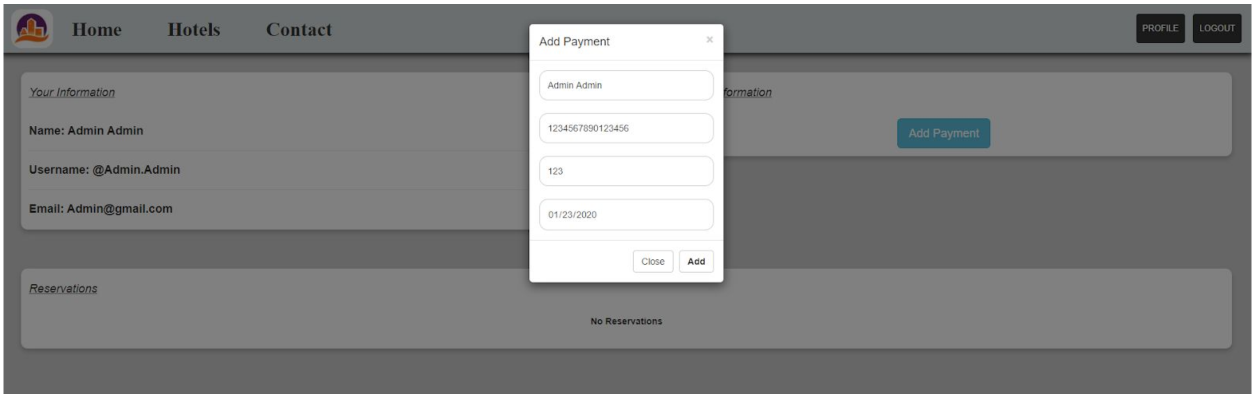
Name: Admin Admin

Username: @Admin.Admin

Email: Admin@gmail.com

Reservations

No Reservations



Add Payment

Admin Admin

1234567890123456

123

01/23/2020

Your Information

Name: Admin Admin

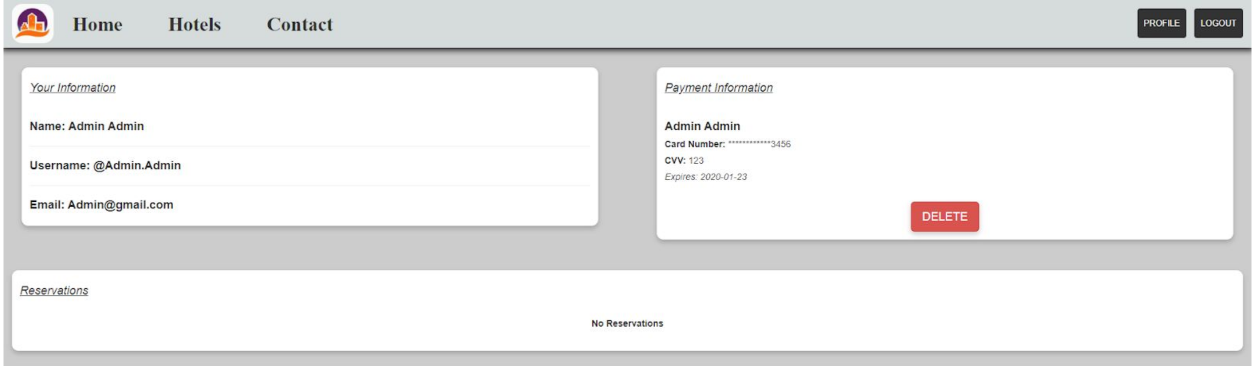
Username: @Admin.Admin

Email: Admin@gmail.com

Reservations

No Reservations

- Saved Payment



Your Information

Name: Admin Admin

Username: @Admin.Admin

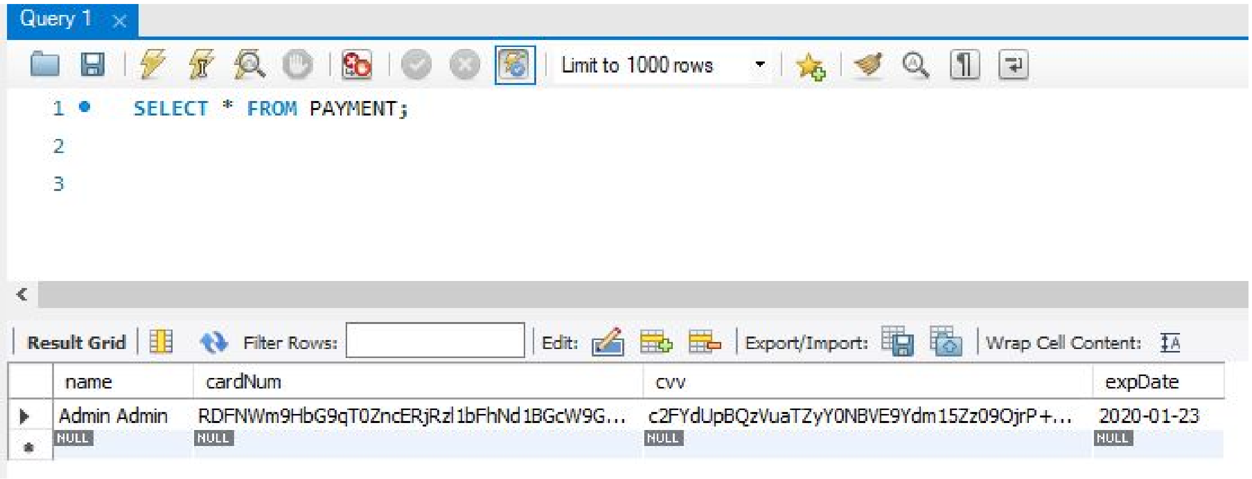
Email: Admin@gmail.com

Payment Information

Admin Admin
Card Number: ************3456
CVV: 123
Expires: 2020-01-23

Reservations

No Reservations



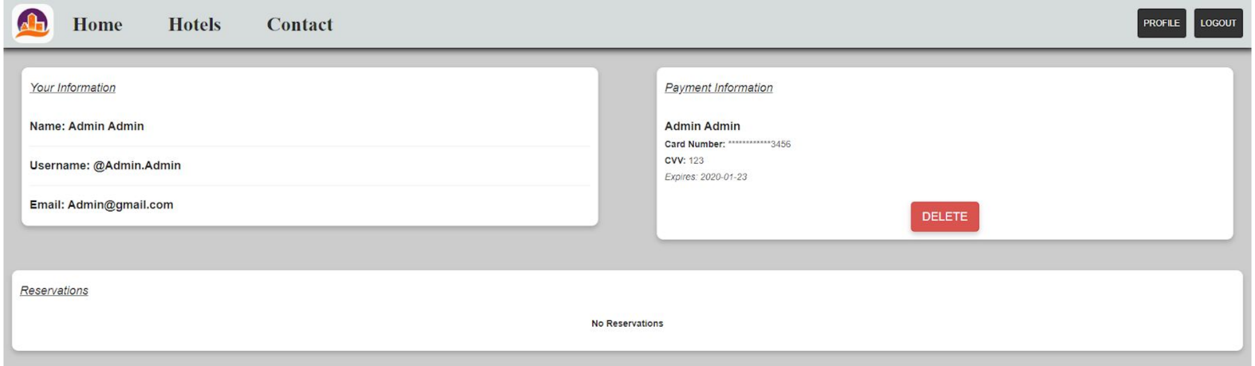
```
SELECT * FROM PAYMENT;
```

name	cardNum	cvv	expDate
Admin Admin	RDFNWm9HbG9qT0ZncERjRzl1bFhNd1BGcW9G...	c2FYdUpBQzVuaTZyY0NBVE9Ydm15Zz09OjrP+...	2020-01-23
NULL	NULL	NULL	NULL

-

8.3.2 Remove Payment

- Delete Option (Red Button)



Your Information

Name: Admin Admin

Username: @Admin.Admin

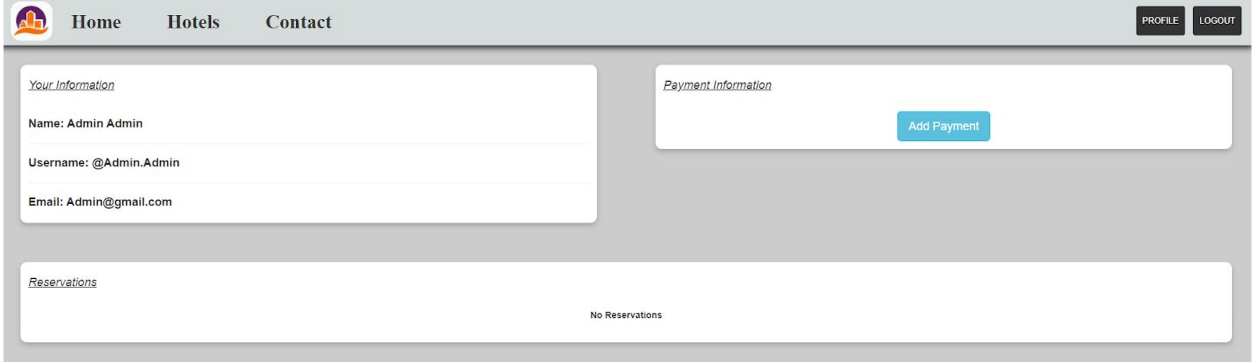
Email: Admin@gmail.com

Payment Information

Admin Admin
Card Number: ************3456
CVV: 123
Expires: 2020-01-23

Reservations

No Reservations



Your Information

Name: Admin Admin

Username: @Admin.Admin

Email: Admin@gmail.com

Payment Information

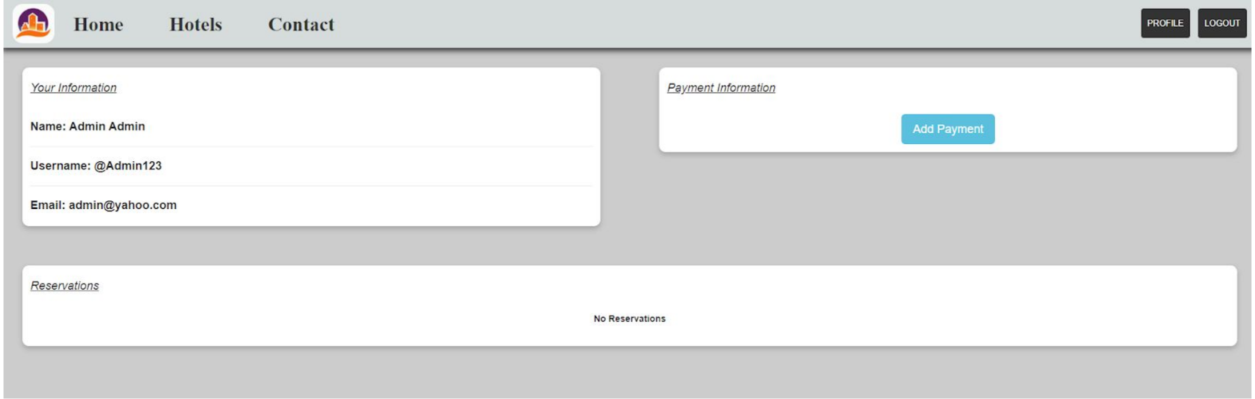
Reservations

No Reservations

8.4 Reserve Room

8.4.1 Reserving Room

- Logged in and on Profile landing page. Note that there is no reservation at the moment



Your Information

Name: Admin Admin

Username: @Admin123

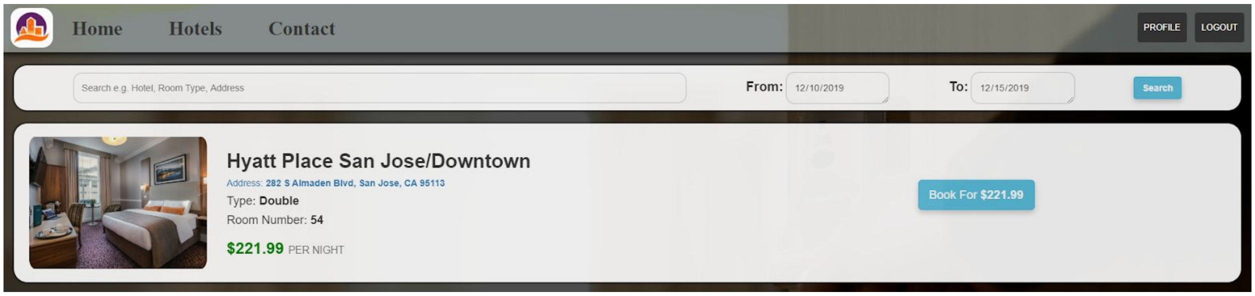
Email: admin@yahoo.com

Payment Information

Reservations

No Reservations

- Home landing page. Select the “Book for $xxx.xx” to make reservation



From: 12/10/2019

To: 12/15/2019

Hyatt Place San Jose/Downtown

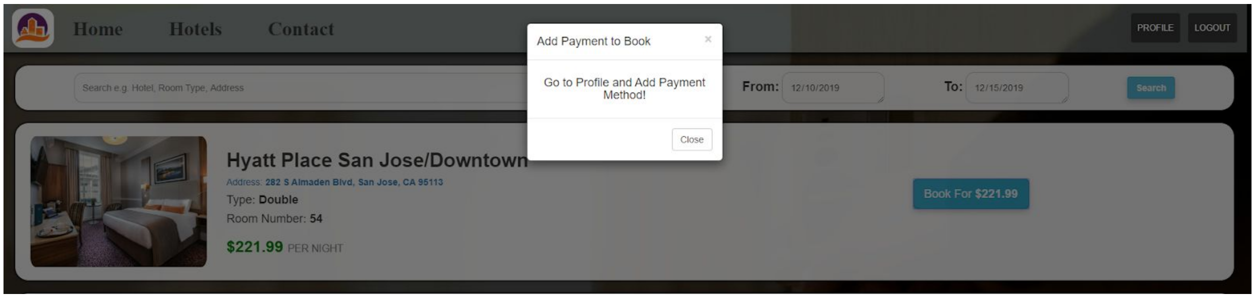
Address: 282 S Almaden Blvd, San Jose, CA 95113

Type: Double

Room Number: 54

$221.99 PER NIGHT

- If payment is not added. To added payment see 8.3.1

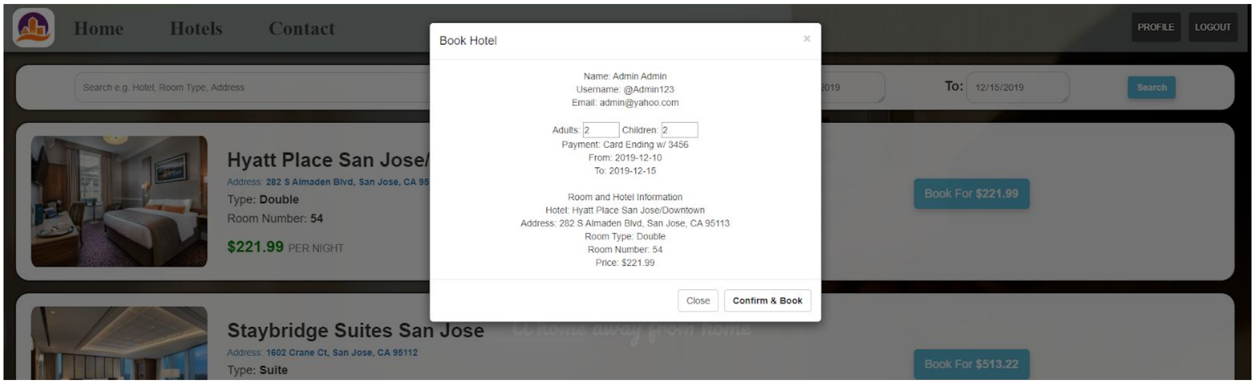


Add Payment to Book

Go to Profile and Add Payment Method!

Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Type: Double
Room Number: 54
$221.99 PER NIGHT

- Booking hotel. Input number of adults and children



Book Hotel

Name: Admin Admin
Username: @Admin123
Email: admin@yahoo.com

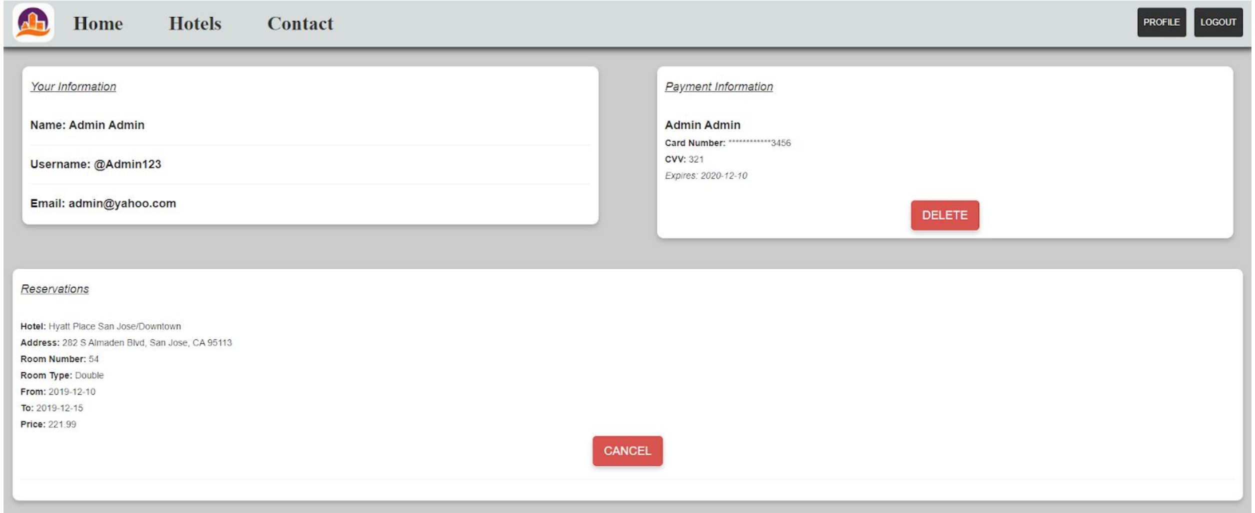
Adults: 2 Children: 2
Payment: Card Ending w/ 3456
From: 2019-12-10
To: 2019-12-15

Room and Hotel Information
Hotel: Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Room Type: Double
Room Number: 54
Price: $221.99

Hyatt Place San Jose/
Address: 282 S Almaden Blvd, San Jose, CA 95
Type: Double
Room Number: 54
$221.99 PER NIGHT

Staybridge Suites San Jose
Address: 1602 Crane Ct, San Jose, CA 95112
Type: Suite

- Visit Profile to confirm that reservation has been made (Bottom of page)



Your Information

Name: Admin Admin

Username: @Admin123

Email: admin@yahoo.com

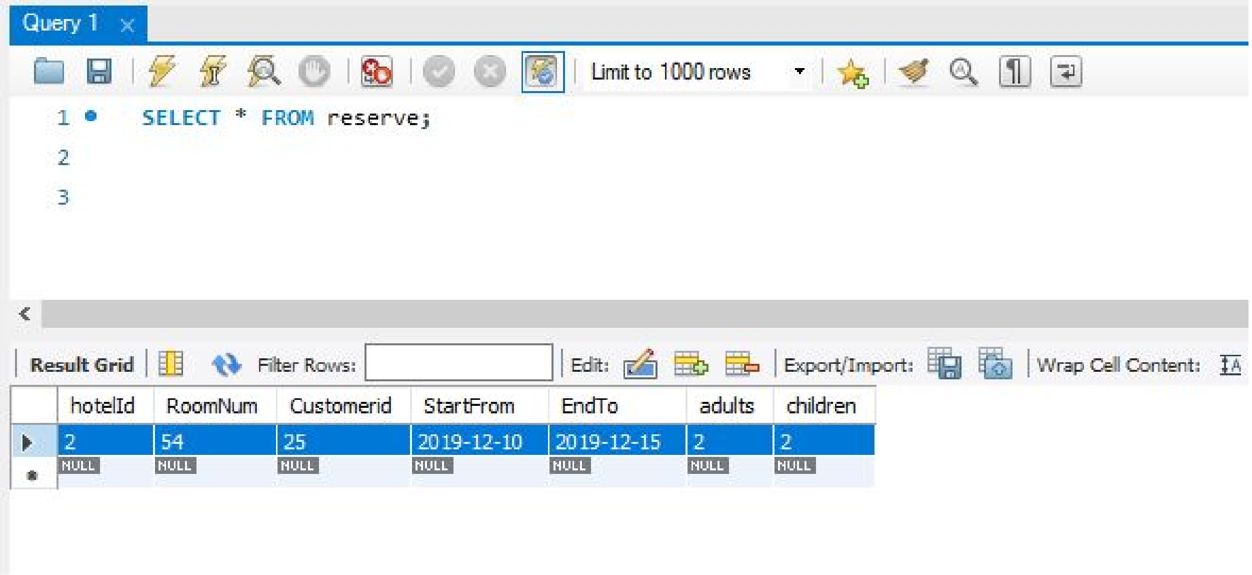
Payment Information

Admin Admin
Card Number: ************3456
CVV: 321
Expires: 2020-12-10

Reservations

Hotel: Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Room Number: 54
Room Type: Double
From: 2019-12-10
To: 2019-12-15
Price: 221.99

- mySQL table showing the reservation

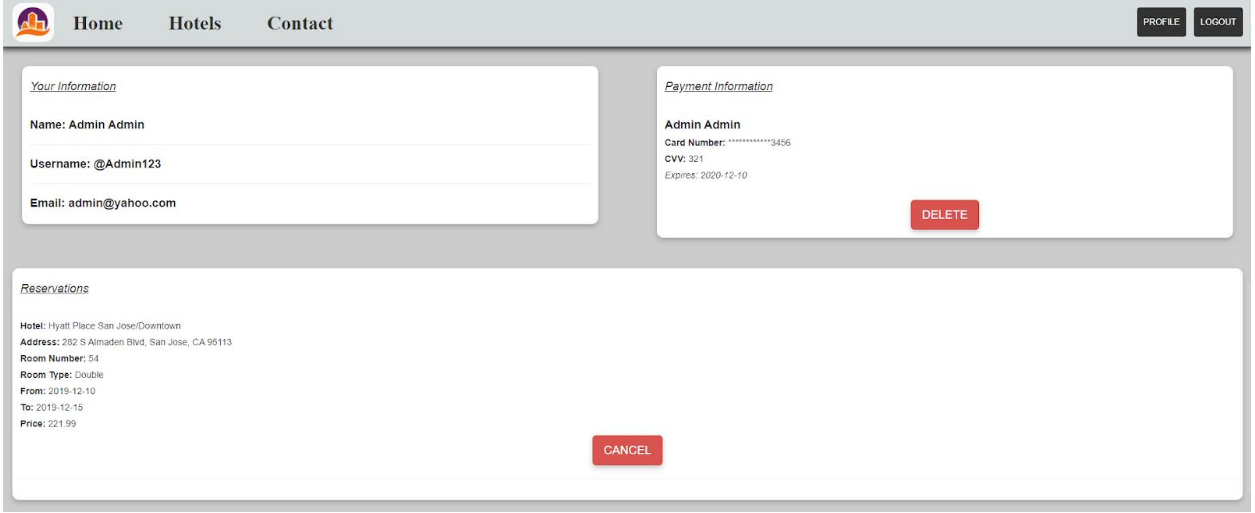


```
SELECT * FROM reserve;
```

hotelId	RoomNum	Customerid	StartFrom	EndTo	adults	children
2	54	25	2019-12-10	2019-12-15	2	2
NULL	NULL	NULL	NULL	NULL	NULL	NULL

8.4.2 Canceling Reservation

- Visit Profile landing page and view reservation at the bottom



Your Information

Name: Admin Admin

Username: @Admin123

Email: admin@yahoo.com

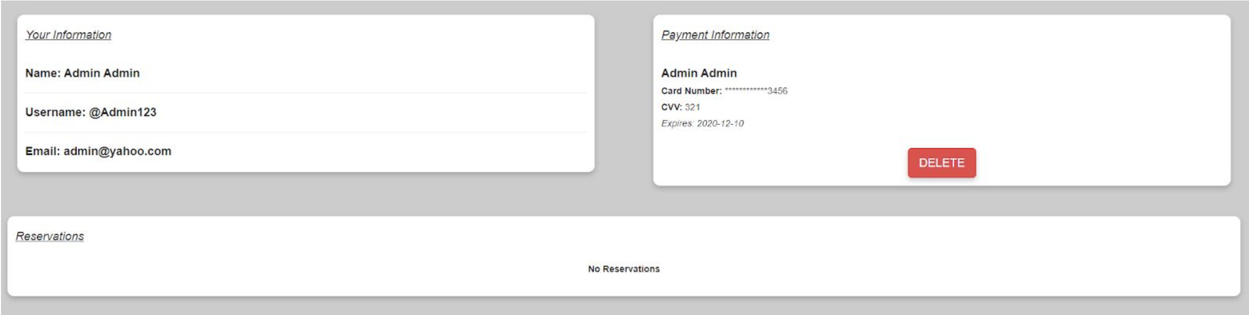
Payment Information

Admin Admin
Card Number: ************3456
CVV: 321
Expires: 2020-12-10

Reservations

Hotel: Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
Room Number: 54
Room Type: Double
From: 2019-12-10
To: 2019-12-15
Price: 221.99

- After pressing ‘CANCEL’



Your Information

Name: Admin Admin

Username: @Admin123

Email: admin@yahoo.com

Payment Information

Admin Admin

Card Number: ************3456

CVV: 321

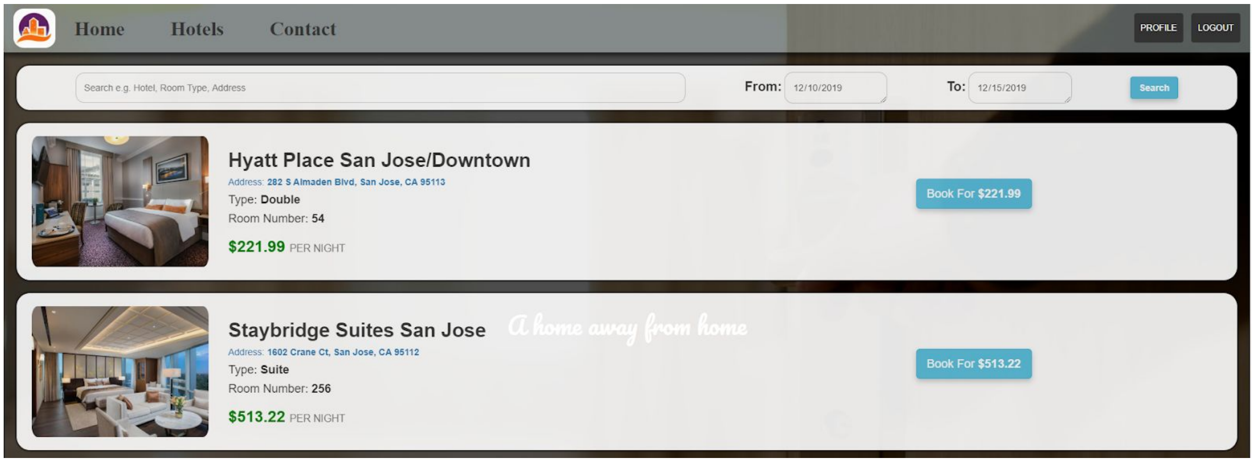
Expires: 2020-12-10

Reservations

No Reservations

8.5 Searching Rooms

- Visit Home landing page



Search e.g. Hotel, Room Type, Address

From: 12/10/2019

To: 12/15/2019

Hyatt Place San Jose/Downtown

Address: 282 S Almaden Blvd, San Jose, CA 95113

Type: Double

Room Number: 54

$221.99 PER NIGHT

Staybridge Suites San Jose

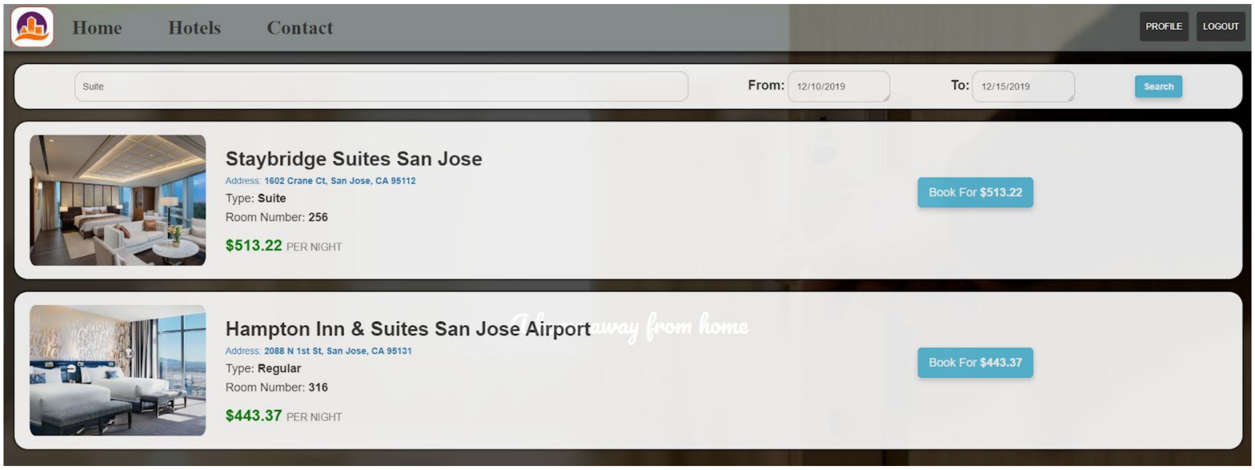
Address: 1602 Crane Ct, San Jose, CA 95112

Type: Suite

Room Number: 256

$513.22 PER NIGHT

- Typing in “Suite” in the search bar narrows down the list of hotels



Suite

From: 12/10/2019

To: 12/15/2019

Staybridge Suites San Jose

Address: 1602 Crane Ct, San Jose, CA 95112

Type: Suite

Room Number: 256

$513.22 PER NIGHT

Hampton Inn & Suites San Jose Airport

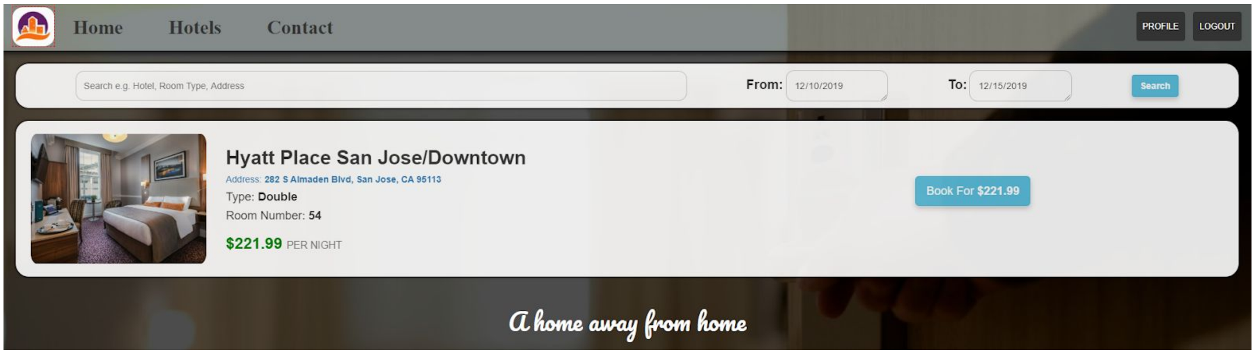
Address: 2088 N 1st St, San Jose, CA 95131

Type: Regular

Room Number: 316

$443.37 PER NIGHT

- Another example searching for “Double”



From: 12/10/2019 To: 12/15/2019

Hyatt Place San Jose/Downtown

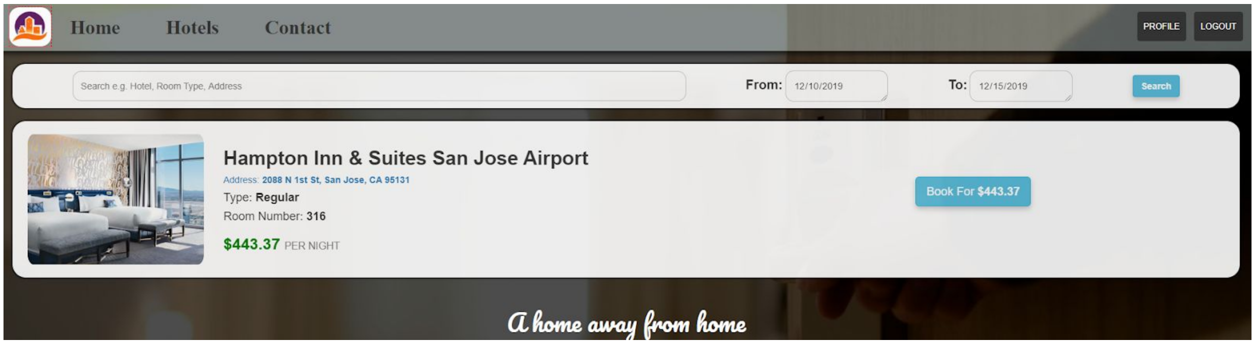
Address: 282 S Almaden Blvd, San Jose, CA 95113

Type: Double

Room Number: 54

$221.99 PER NIGHT

- Hotels can also be searched by their name. This example searches for “Hampton”



From: 12/10/2019 To: 12/15/2019

Hampton Inn & Suites San Jose Airport

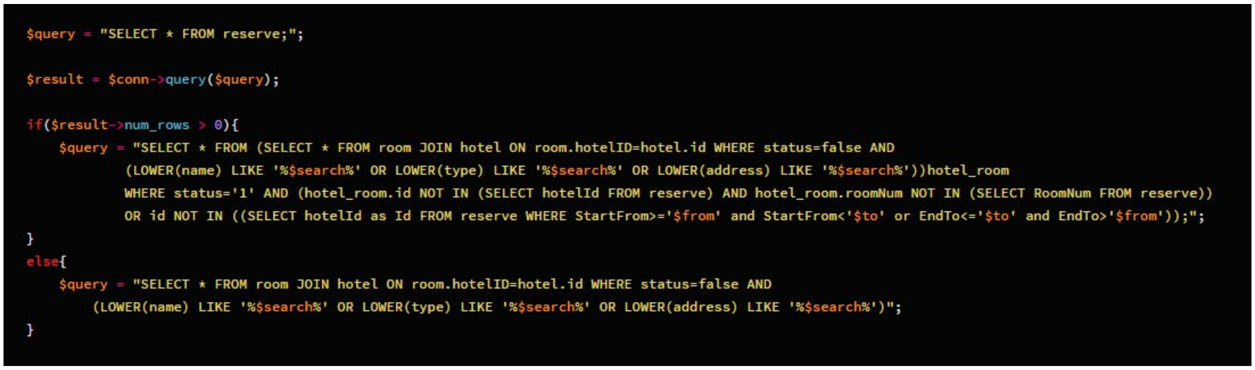
Address: 2088 N 1st St, San Jose, CA 95131

Type: Regular

Room Number: 316

$443.37 PER NIGHT

- SQL statement for search



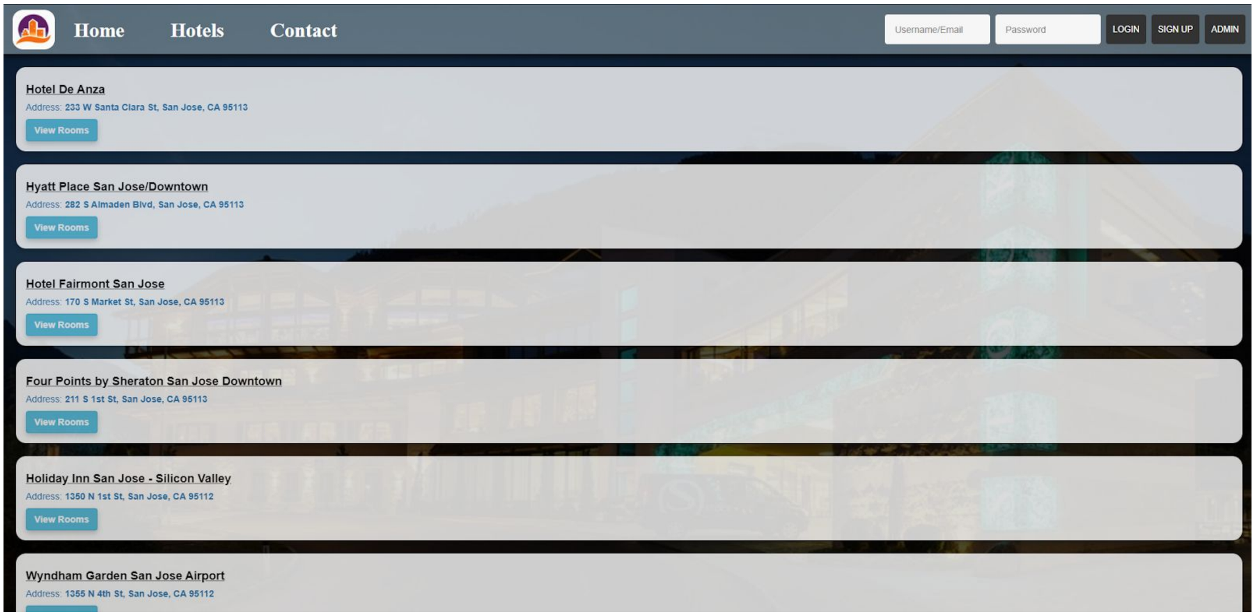
```
$query = "SELECT * FROM reserve;";

$result = $conn->query($query);

if($result->num_rows > 0){
    $query = "SELECT * FROM (SELECT * FROM room JOIN hotel ON room.hotelID=hotel.id WHERE status=false AND
            (LOWER(name) LIKE '%$search%' OR LOWER(type) LIKE '%$search%' OR LOWER(address) LIKE '%$search%'))hotel_room
            WHERE status='1' AND (hotel_room.id NOT IN (SELECT hotelId FROM reserve) AND hotel_room.roomNum NOT IN (SELECT RoomNum FROM reserve))
            OR id NOT IN ((SELECT hotelId as Id FROM reserve WHERE StartFrom>='$from' and StartFrom<'$to' or EndTo<='$to' and EndTo>'$from'));";
}
else{
    $query = "SELECT * FROM room JOIN hotel ON room.hotelID=hotel.id WHERE status=false AND
        (LOWER(name) LIKE '%$search%' OR LOWER(type) LIKE '%$search%' OR LOWER(address) LIKE '%$search%')";
}
```

8.6 Viewing Hotels

- Hotel



Hotel De Anza
Address: 233 W Santa Clara St, San Jose, CA 95113
View Rooms

Hyatt Place San Jose/Downtown
Address: 282 S Almaden Blvd, San Jose, CA 95113
View Rooms

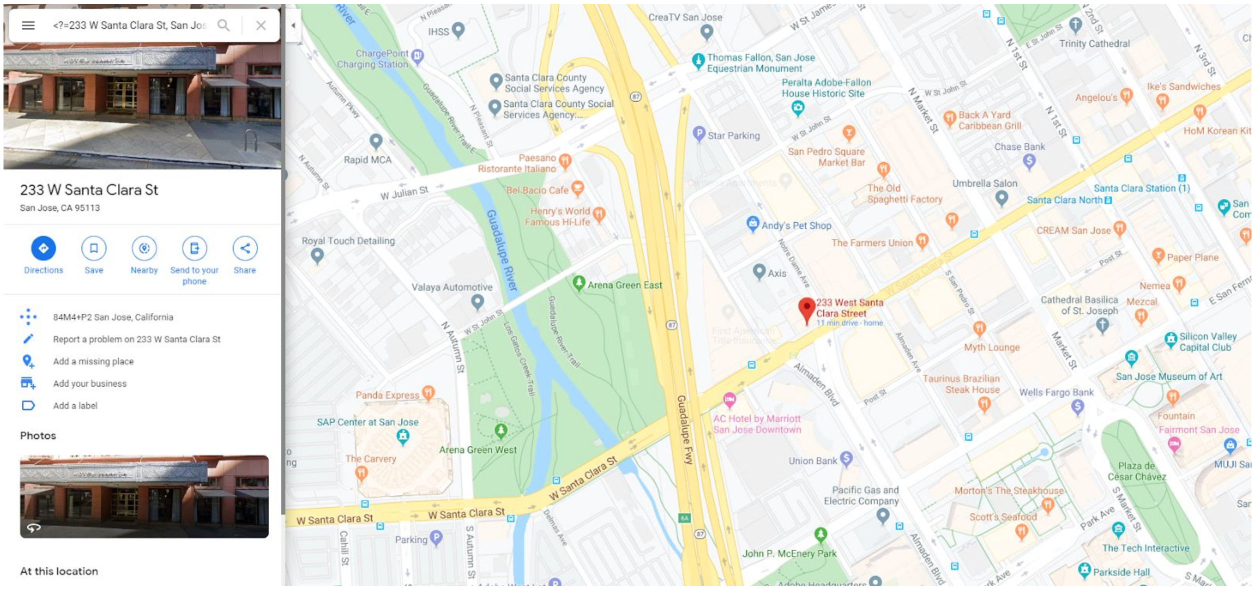
Hotel Fairmont San Jose
Address: 170 S Market St, San Jose, CA 95113
View Rooms

Four Points by Sheraton San Jose Downtown
Address: 211 S 1st St, San Jose, CA 95113
View Rooms

Holiday Inn San Jose - Silicon Valley
Address: 1350 N 1st St, San Jose, CA 95112
View Rooms

Wyndham Garden San Jose Airport
Address: 1355 N 4th St, San Jose, CA 95112

- Selecting the address of Hotel De Anza



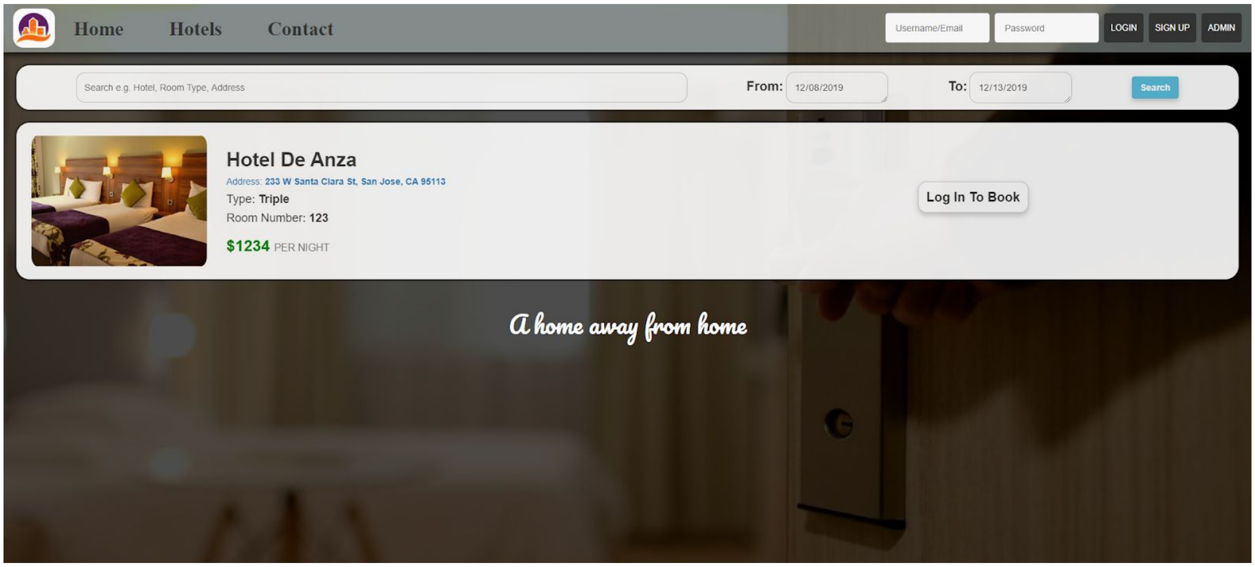
233 W Santa Clara St
San Jose, CA 95113

84M4+P2 San Jose, California
Report a problem on 233 W Santa Clara St
Add a missing place
Add your business
Add a label

Photos

At this location

- Selecting View Rooms of Hotel De Anza



From: 12/08/2019 To: 12/13/2019

Hotel De Anza

Address: 233 W Santa Clara St, San Jose, CA 95113

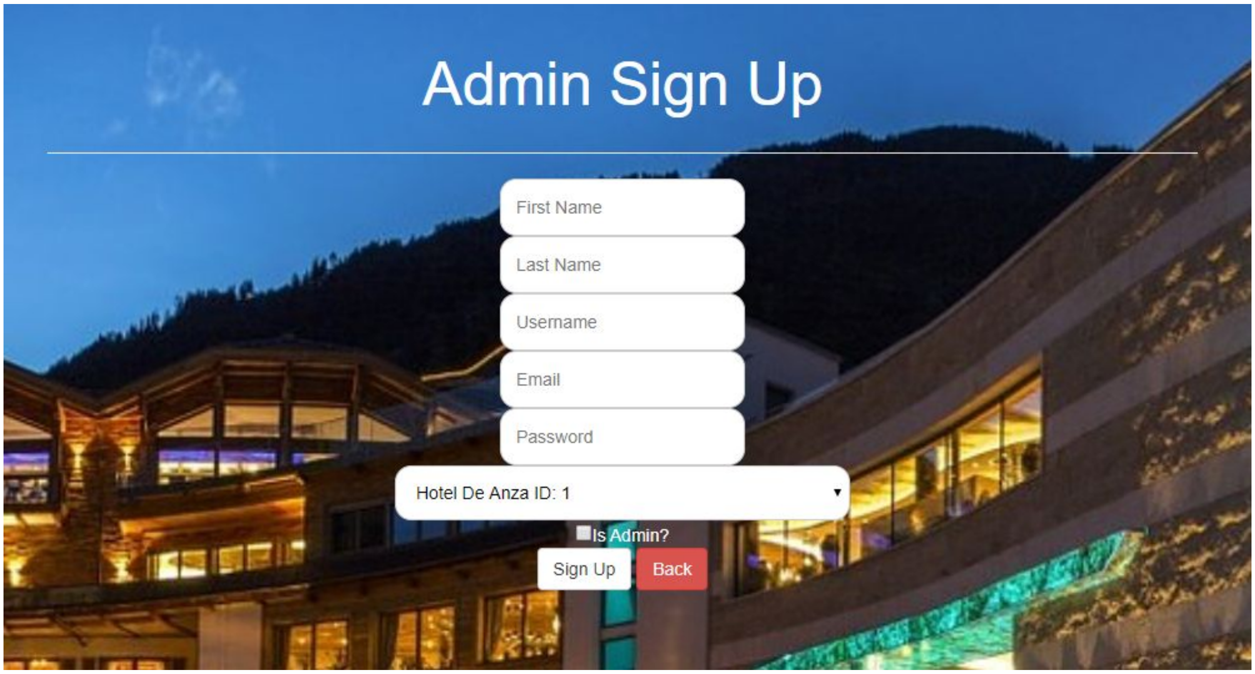
Type: Triple

Room Number: 123

$1234 PER NIGHT

8.7 Admin Sign Up

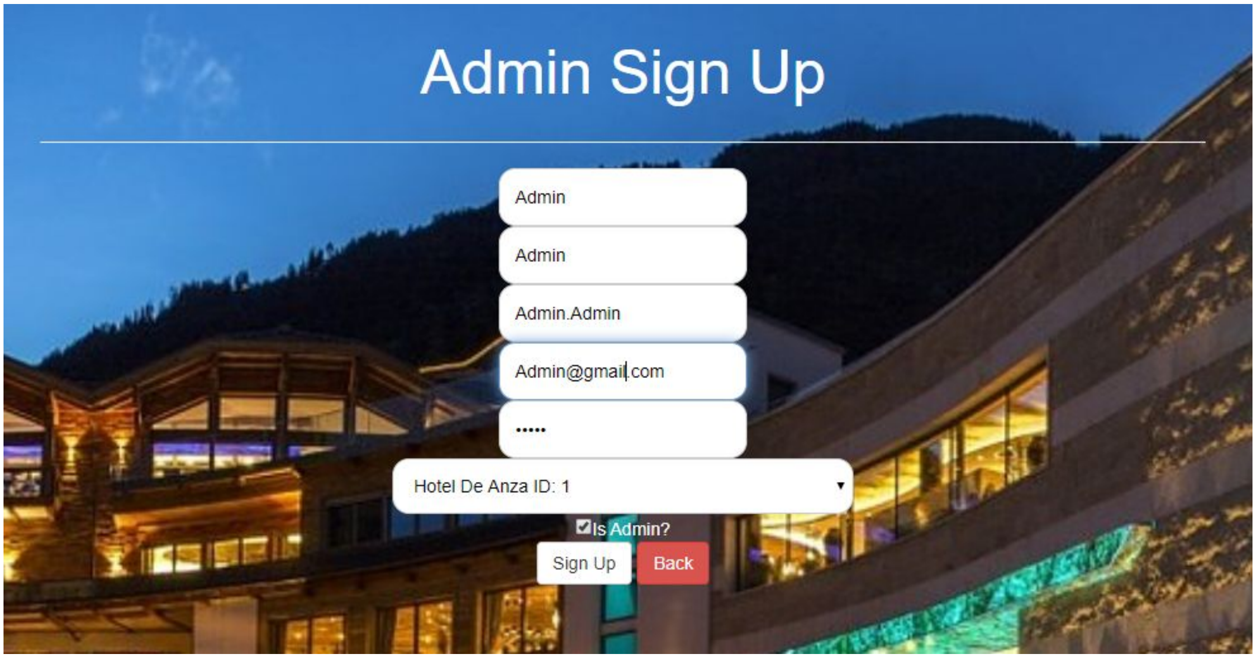
- Landing Page



Hotel De Anza ID: 1 ▾

□ Is Admin?

- Information



Admin Sign Up

Admin

Admin

Admin.Admin

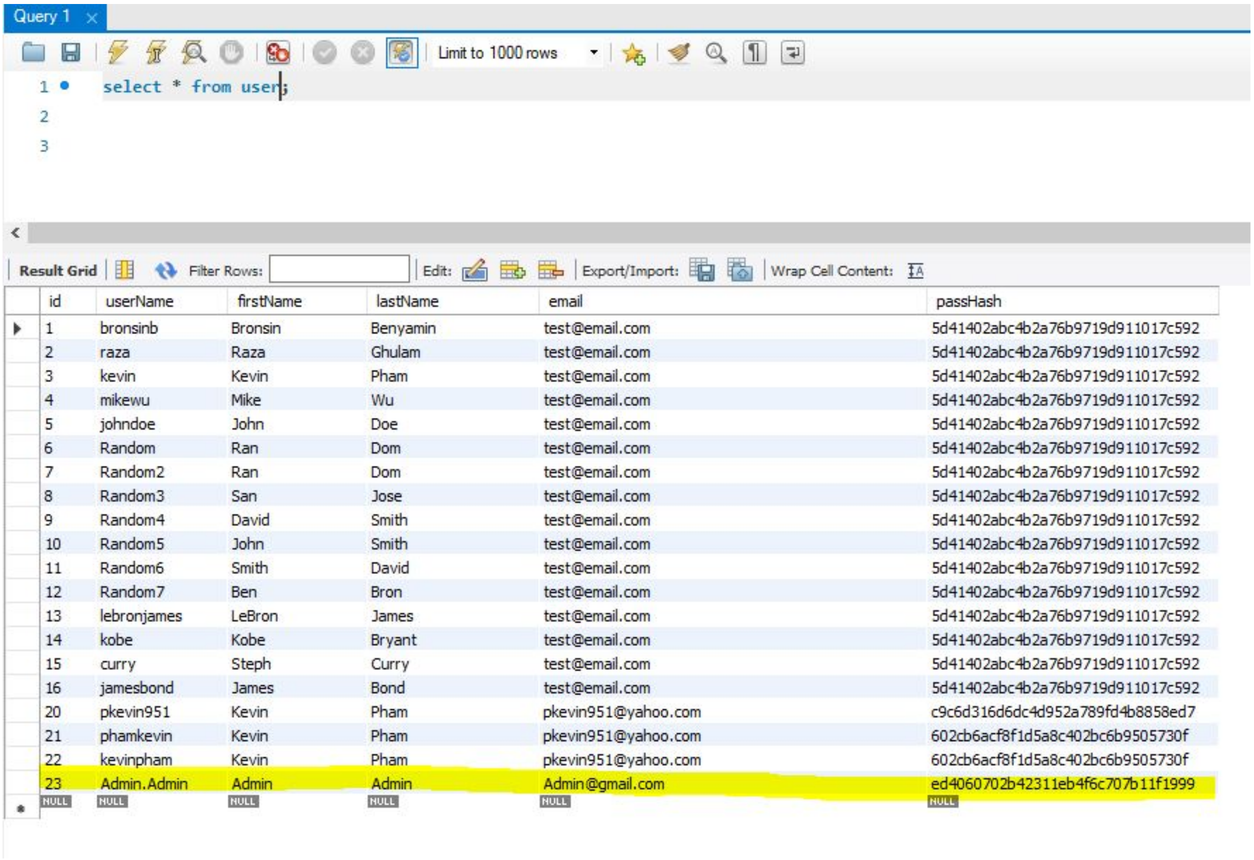
Admin@gmail.com

•••••

Hotel De Anza ID: 1 ▾

☑ Is Admin?

- MySQL Result

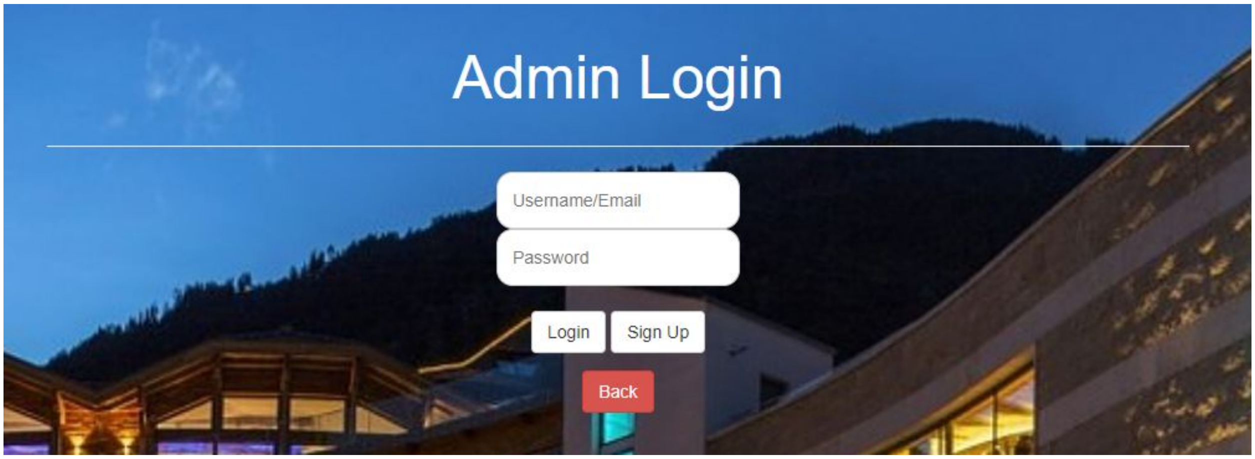


```
select * from user;
```

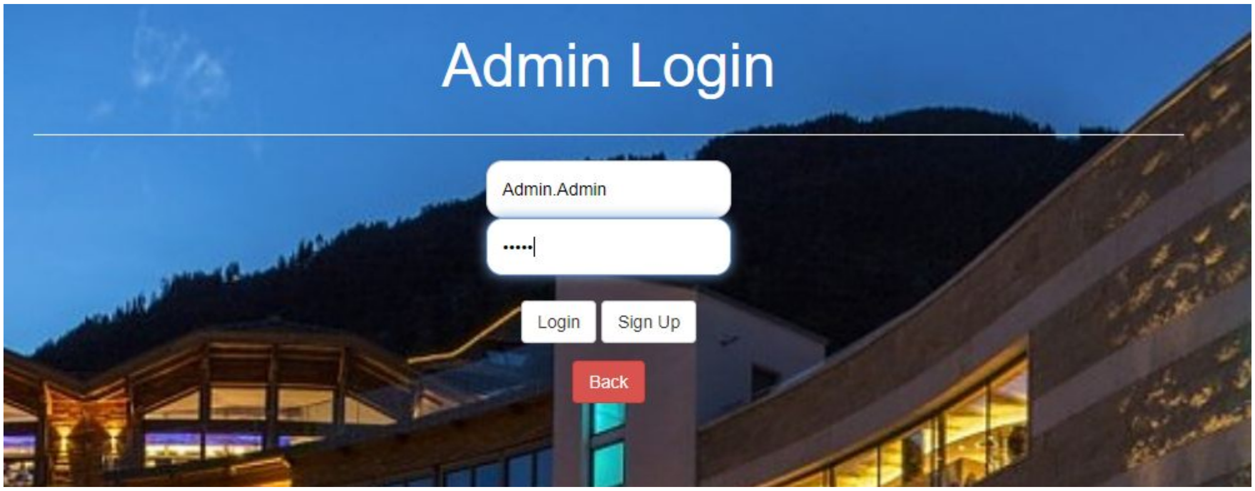
id	userName	firstName	lastName	email	passHash
1	bronsinb	Bronsin	Benyamin	test@email.com	5d41402abc4b2a76b9719d911017c592
2	raza	Raza	Ghulam	test@email.com	5d41402abc4b2a76b9719d911017c592
3	kevin	Kevin	Pham	test@email.com	5d41402abc4b2a76b9719d911017c592
4	mikewu	Mike	Wu	test@email.com	5d41402abc4b2a76b9719d911017c592
5	johndoe	John	Doe	test@email.com	5d41402abc4b2a76b9719d911017c592
6	Random	Ran	Dom	test@email.com	5d41402abc4b2a76b9719d911017c592
7	Random2	Ran	Dom	test@email.com	5d41402abc4b2a76b9719d911017c592
8	Random3	San	Jose	test@email.com	5d41402abc4b2a76b9719d911017c592
9	Random4	David	Smith	test@email.com	5d41402abc4b2a76b9719d911017c592
10	Random5	John	Smith	test@email.com	5d41402abc4b2a76b9719d911017c592
11	Random6	Smith	David	test@email.com	5d41402abc4b2a76b9719d911017c592
12	Random7	Ben	Bron	test@email.com	5d41402abc4b2a76b9719d911017c592
13	lebronjames	LeBron	James	test@email.com	5d41402abc4b2a76b9719d911017c592
14	kobe	Kobe	Bryant	test@email.com	5d41402abc4b2a76b9719d911017c592
15	curry	Steph	Curry	test@email.com	5d41402abc4b2a76b9719d911017c592
16	jamesbond	James	Bond	test@email.com	5d41402abc4b2a76b9719d911017c592
20	pkevin951	Kevin	Pham	pkevin951@yahoo.com	c9c6d316d6dc4d952a789fd4b8858ed7
21	phamkevin	Kevin	Pham	pkevin951@yahoo.com	602cb6acf8f1d5a8c402bc6b9505730f
22	kevinpham	Kevin	Pham	pkevin951@yahoo.com	602cb6acf8f1d5a8c402bc6b9505730f
23	Admin.Admin	Admin	Admin	Admin@gmail.com	ed4060702b42311eb4f6c707b11f1999
NULL	NULL	NULL	NULL	NULL	NULL

8.8 Admin Login

- Login Landing Page

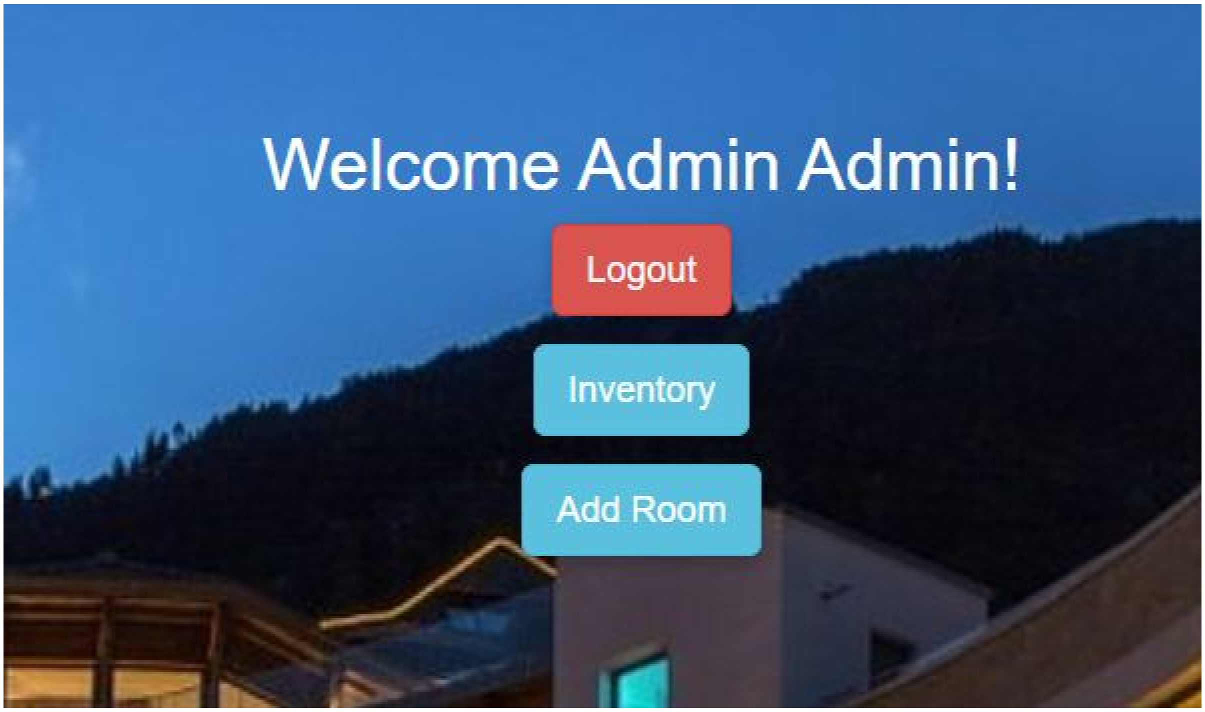


- Filling out with information above



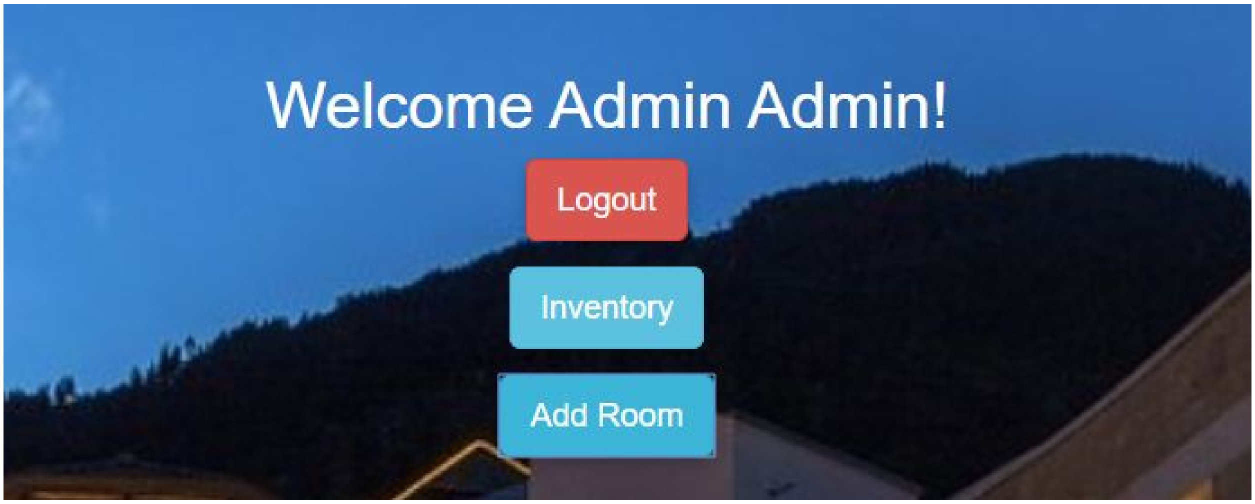
Admin.Admin

•••••

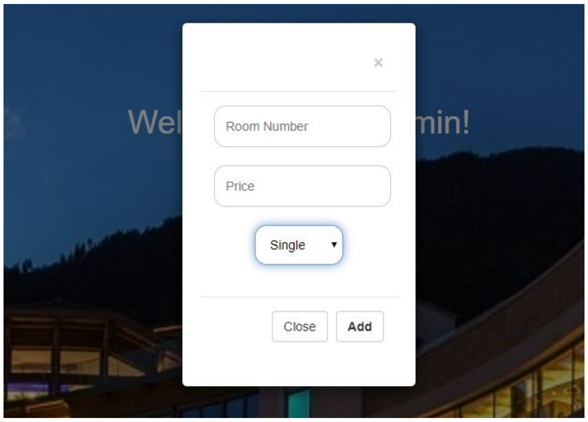


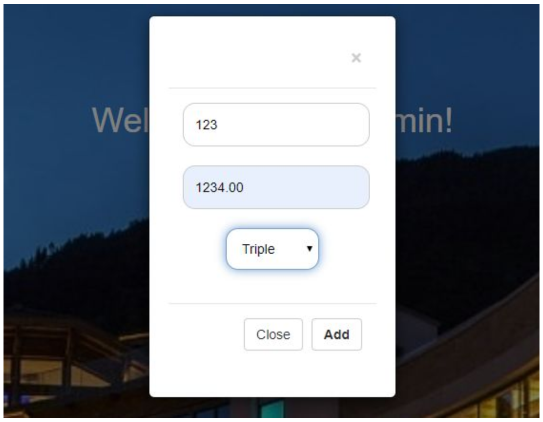
8.9 Admin Room Editing

- Initial landing page when logged into Admin



- ‘Add Room’



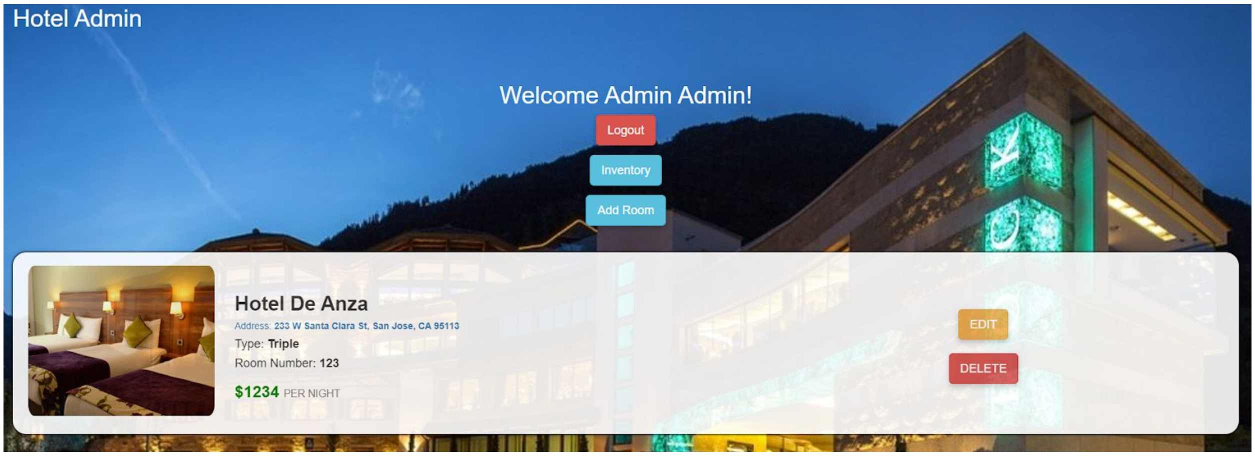


123

1234.00

Triple

- Newly added room



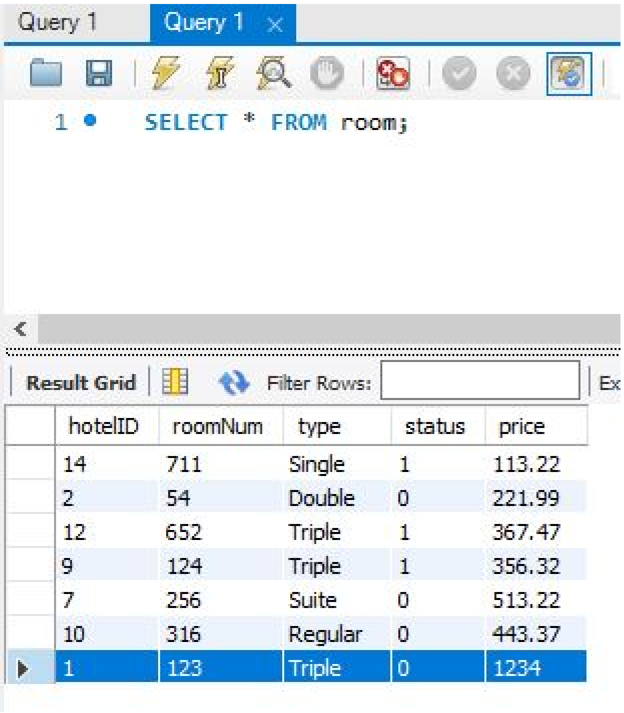
Hotel De Anza

Address: 233 W Santa Clara St, San Jose, CA 95113

Type: Triple

Room Number: 123

$1234 PER NIGHT



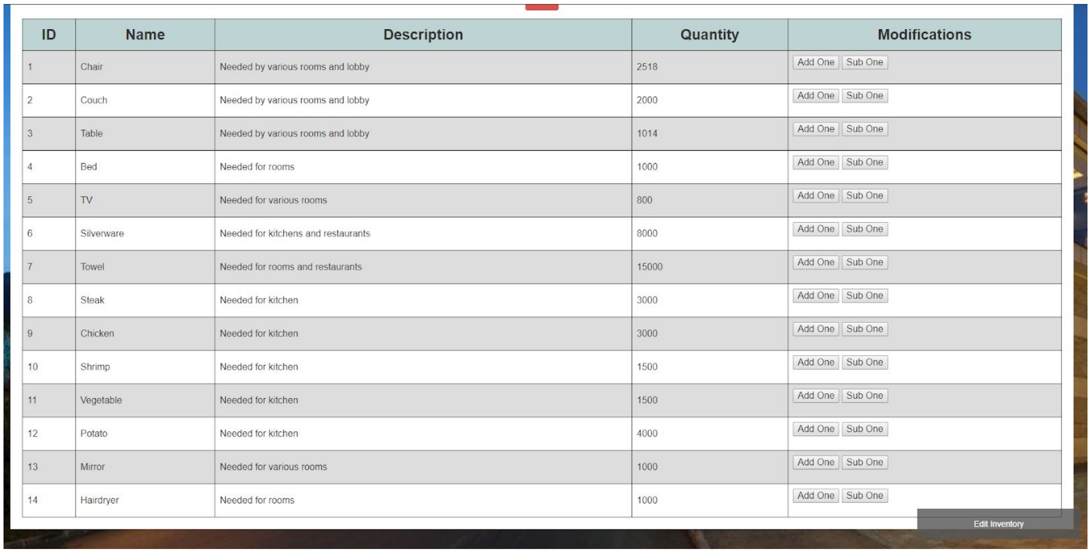
```
SELECT * FROM room;
```

hotelID	roomNum	type	status	price
14	711	Single	1	113.22
2	54	Double	0	221.99
12	652	Triple	1	367.47
9	124	Triple	1	356.32
7	256	Suite	0	513.22
10	316	Regular	0	443.37
1	123	Triple	0	1234

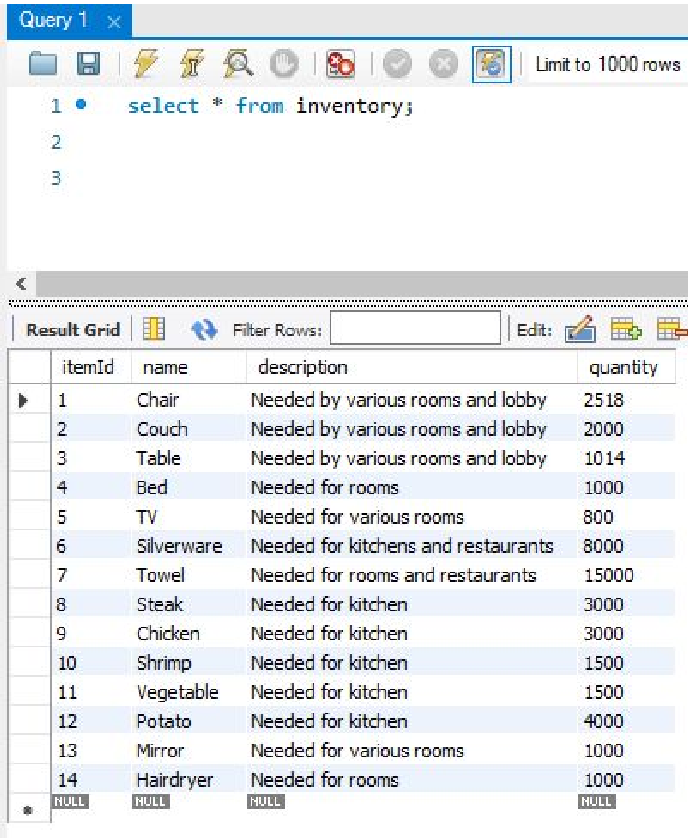
8.10 Admin Inventory

8.10.1 Adding to Inventory

- Beginning inventory



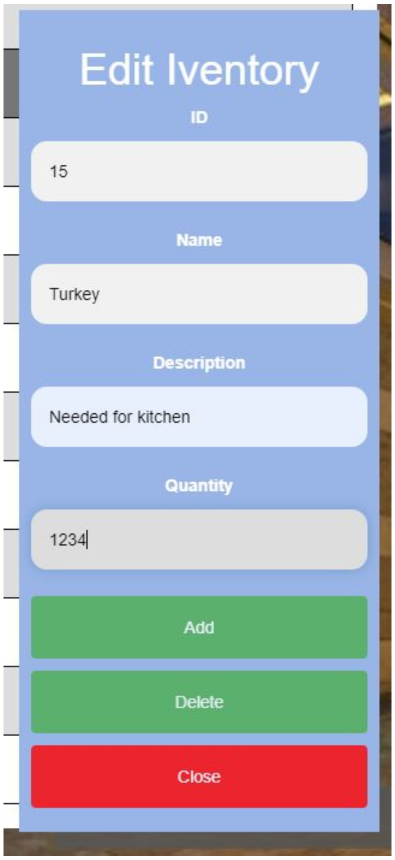
ID	Name	Description	Quantity	Modifications
1	Chair	Needed by various rooms and lobby	2518	Add One Sub One
2	Couch	Needed by various rooms and lobby	2000	Add One Sub One
3	Table	Needed by various rooms and lobby	1014	Add One Sub One
4	Bed	Needed for rooms	1000	Add One Sub One
5	TV	Needed for various rooms	800	Add One Sub One
6	Silverware	Needed for kitchens and restaurants	8000	Add One Sub One
7	Towel	Needed for rooms and restaurants	15000	Add One Sub One
8	Steak	Needed for kitchen	3000	Add One Sub One
9	Chicken	Needed for kitchen	3000	Add One Sub One
10	Shrimp	Needed for kitchen	1500	Add One Sub One
11	Vegetable	Needed for kitchen	1500	Add One Sub One
12	Potato	Needed for kitchen	4000	Add One Sub One
13	Mirror	Needed for various rooms	1000	Add One Sub One
14	Hairdryer	Needed for rooms	1000	Add One Sub One



```
select * from inventory;
```

itemId	name	description	quantity
1	Chair	Needed by various rooms and lobby	2518
2	Couch	Needed by various rooms and lobby	2000
3	Table	Needed by various rooms and lobby	1014
4	Bed	Needed for rooms	1000
5	TV	Needed for various rooms	800
6	Silverware	Needed for kitchens and restaurants	8000
7	Towel	Needed for rooms and restaurants	15000
8	Steak	Needed for kitchen	3000
9	Chicken	Needed for kitchen	3000
10	Shrimp	Needed for kitchen	1500
11	Vegetable	Needed for kitchen	1500
12	Potato	Needed for kitchen	4000
13	Mirror	Needed for various rooms	1000
14	Hairdryer	Needed for rooms	1000
NULL	NULL	NULL	NULL

- Adding ‘Turkey’ to inventory list



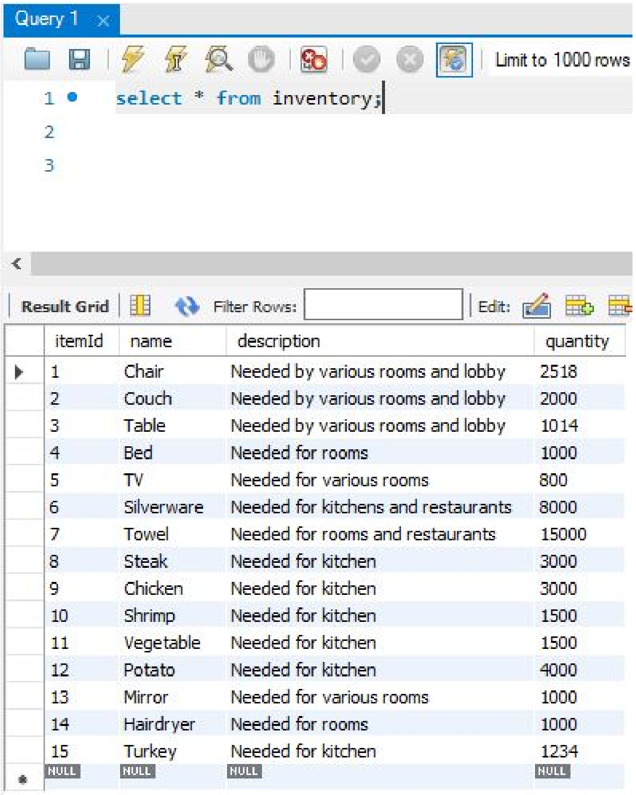
Edit Iventory

ID: 15

Name: Turkey

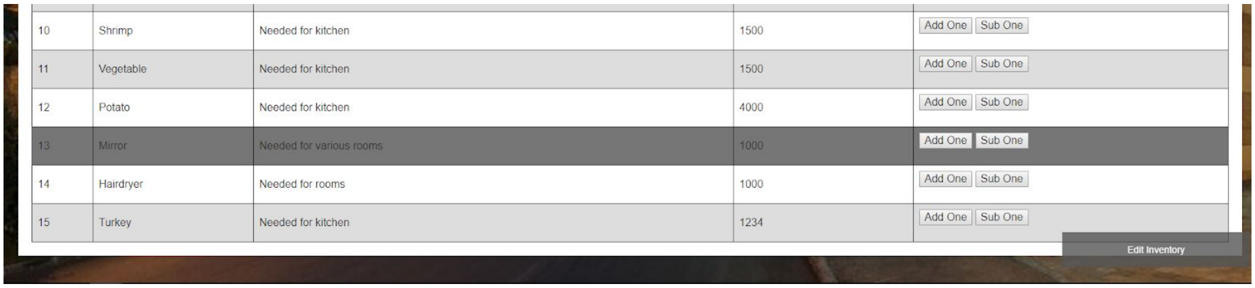
Description: Needed for kitchen

Quantity: 1234



```
select * from inventory;
```

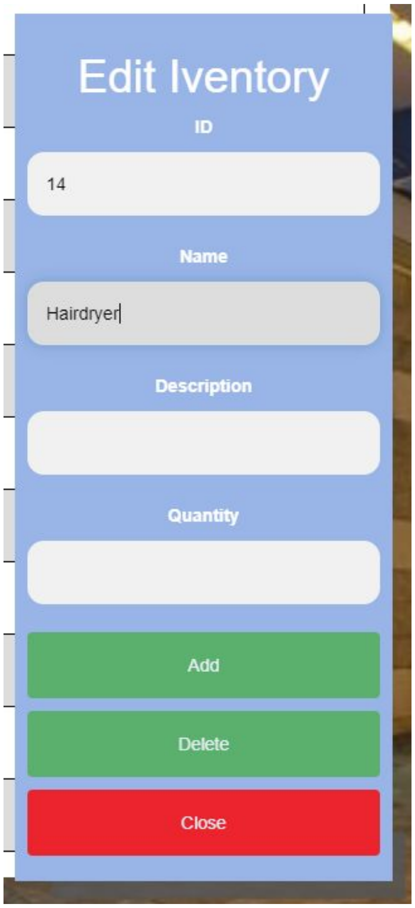
itemId	name	description	quantity
1	Chair	Needed by various rooms and lobby	2518
2	Couch	Needed by various rooms and lobby	2000
3	Table	Needed by various rooms and lobby	1014
4	Bed	Needed for rooms	1000
5	TV	Needed for various rooms	800
6	Silverware	Needed for kitchens and restaurants	8000
7	Towel	Needed for rooms and restaurants	15000
8	Steak	Needed for kitchen	3000
9	Chicken	Needed for kitchen	3000
10	Shrimp	Needed for kitchen	1500
11	Vegetable	Needed for kitchen	1500
12	Potato	Needed for kitchen	4000
13	Mirror	Needed for various rooms	1000
14	Hairdryer	Needed for rooms	1000
15	Turkey	Needed for kitchen	1234
NULL	NULL	NULL	NULL



10	Shrimp	Needed for kitchen	1500	Add One Sub One
11	Vegetable	Needed for kitchen	1500	Add One Sub One
12	Potato	Needed for kitchen	4000	Add One Sub One
13	Mirror	Needed for various rooms	1000	Add One Sub One
14	Hairdryer	Needed for rooms	1000	Add One Sub One
15	Turkey	Needed for kitchen	1234	Add One Sub One

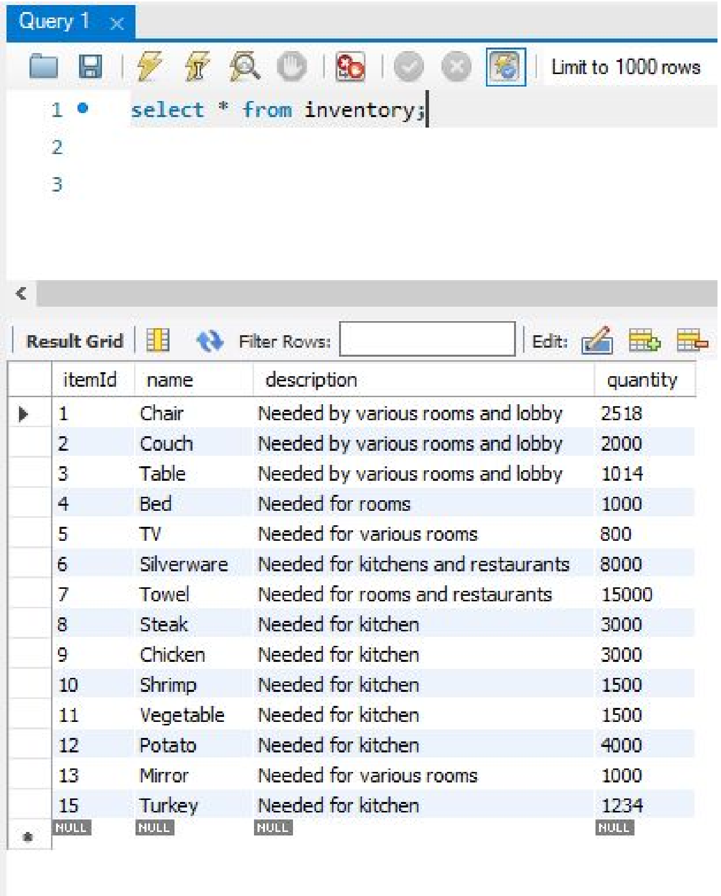
8.10.2 Removing from Inventory

- In this example, we remove Hair Dryer (ID 14)



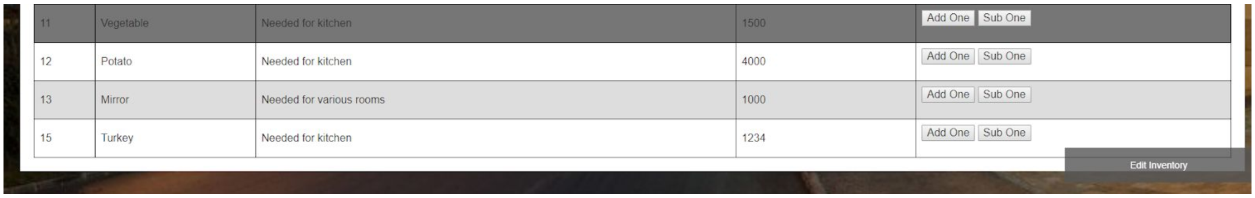
Edit Iventory

ID	14
Name	Hairdryer
Description	
Quantity	



```
select * from inventory;
```

itemId	name	description	quantity
1	Chair	Needed by various rooms and lobby	2518
2	Couch	Needed by various rooms and lobby	2000
3	Table	Needed by various rooms and lobby	1014
4	Bed	Needed for rooms	1000
5	TV	Needed for various rooms	800
6	Silverware	Needed for kitchens and restaurants	8000
7	Towel	Needed for rooms and restaurants	15000
8	Steak	Needed for kitchen	3000
9	Chicken	Needed for kitchen	3000
10	Shrimp	Needed for kitchen	1500
11	Vegetable	Needed for kitchen	1500
12	Potato	Needed for kitchen	4000
13	Mirror	Needed for various rooms	1000
15	Turkey	Needed for kitchen	1234
NULL	NULL	NULL	NULL



11	Vegetable	Needed for kitchen	1500	Add One Sub One
12	Potato	Needed for kitchen	4000	Add One Sub One
13	Mirror	Needed for various rooms	1000	Add One Sub One
15	Turkey	Needed for kitchen	1234	Add One Sub One

9.0 Project Conclusion

9.1 Lessons Learned

9.1.1 Bronsin Benyamin Pouran

Before this project, I had some experience developing a web application using MySQL, PHP, HTML, CSS and Apache Web Server. However, I did not know anything about how to design a relational database and the powerful commands MySQL has for querying such as subquerying, GROUP BY, NOT IN etc. With this DB project, I learned to design a relational database using ER diagrams, functional dependencies, and normalization. The ER diagram is a great because it shows the structure your database and how your tables will relate and communicate with each other. I will definitely create ER diagrams whenever starting a new project. I also learned different powerful MYSQL commands to use in the project to make features like searching and displaying rooms based on the search and reservations. These commands made the searching in our HMS website very powerful. With four lines of code, MySQL displayed rooms based on address, hotel name, room type and date range. The knowledge I gained from this project will definitely help me out in the future.

9.1.2 Kevin Pham

I would like to start off by saying this is one of my more enjoyable classes this semester in terms of learning experience. Going into the course, I have zero knowledge of database manipulation, mySQL, PHP, and very little experience with HTML and CSS. When we were first introduced with this assignment, I was slightly concerned as I have done nothing of this level in terms of frontend and backend development. However, as the course went by, we learn many techniques that we use for database manipulation. All I had to learn a little on the side was HTML, CSS, PHP and combine them all together. What I enjoyed about this project is that it is comparable to a "real life" application. It looks realistic. Its functionality is realistic. Next semester, I am taking a Software Engineering II class that requires us to develop some kind of application. I now can confidently go into that class knowing that I can use my experience with this project to help me. Overall, I am proud to say that I was a part of this beautiful project and a wonderful class. Not to mention, I also got to meet two great friends along the way!

9.1.3 Raza Ghulam

The only thing I knew prior to this class was database system definition. Prior to this class, I had no knowledge of HTML, PHP, CSS and MySql. As the class started with introducing the concept of database and three-tier structure, I found myself in the middle of nowhere. It was the second week that I decided to spend more time to prepare myself for this class and I did. I started to learn HTML followed by PHP and a little bit of CSS. I was in the process of learning these languages that the project was assigned. As a team, we decided to create a Hotel Management System. I found our first team meeting inspiring. During our first team meeting session, I felt that we were working for some company. After discussing the project details and completing the documentation, we moved forward and designed our E/R diagram, created our relations, and implemented our functional requirements. Through this realistic project, I think I am much more prepared and have the ability to put hands on real world realistic projects. I also want to mention that I really enjoyed working with Bronsin and Kevin. Both were talented, polite and hardworking and without them, I would never finish this project. In conclusion, through this project, not only have I learned database design and implementation but have also improved my teamwork skills and communication.

9.2 Future Improvements

Our Hotel Management System application is a decent application that provides basic functionalities but there is still room for improvements and features. There are functionalities that can be added in the future to make it more functional. Adding the following functionalities will help us to gain more user and hotel staff satisfaction:

- Allow employees to book rooms for themselves or relatives
- Allow admins to edit employee information such as salary
- Allow admins and employees to see their information such as salary
- Allow admins and certain employees to view reservations on their hotel's rooms and view other relevant data such as profits.
- Allow customers to choose the type of credit card they want to use
- Allow customers to use more than one card to pay for a single transaction.
- Offer coupons to customers and college students
- Offer discount to military personnel and veterans
- Allow customers to give feedback.
- Add cancelation policy