**INFORMATION ABOUT MY WEBSITE**

**SOURCE FOR LOGO:**

Vacation Inn’s logowas made with Logomakr.com (https://logomakr.com)

**SOURCES FOR ALL IMAGES USED ON WEBSITE:**

Master Bedroom Image: <https://pixabay.com/en/master-bedroom-residence-home-house-2014865/>

Double Bedroom Image: <https://pixabay.com/en/travel-hotel-room-hotel-room-bed-1550377/>

Standard Room Image: <https://pixabay.com/en/hotel-podjavorn%C3%ADk-room-1191709/>

Hotel Conference/Dining Room Image: <https://pixabay.com/en/interior-hotel-rendering-1026456/>

Hotel Exterior Image: <https://pixabay.com/en/roznowskie-lake-pictures-of-drone-1273941/>

Hotel Beach Image: <https://pixabay.com/en/blue-sky-white-cloud-beach-tourism-2433887/>

Hotel Pool Image: <https://pixabay.com/en/pool-cottage-interior-1318072/>

Hotel Restaurant Image: <https://pixabay.com/en/hotel-berlin-kurf%C3%BCrstendamm-city-951599/>

**REGISTERED DATABASE USERS (Customer-level access):**

|  |  |  |  |
| --- | --- | --- | --- |
| **Email** | **Password** | **Name** | **Inspiration for User** |
| [fake@yahoo.com](mailto:fake@yahoo.com) | yyyyy | Diana Poelvoorde | My mother |
| [fast@gmail.com](mailto:fast@gmail.com) | tails123 | Sonic Hedgehog | Video game character from early 1990s |
| [fake@blah.com](mailto:fake@blah.com) | sonic123 | Miles Prowler | “Tails” from the *Sonic* video games |

**REGISTERED DATABASE USERS (Managerial-level access):**

|  |  |  |  |
| --- | --- | --- | --- |
| **Email** | **Password** | **Name** | **Inspiration for User** |
| [fake@gmail.com](mailto:fake@gmail.com) | xxxxx | Mathew Poelvoorde | Myself |
| [slow@gmail.com](mailto:slow@gmail.com) | eggman456 | Knuckles Echidna | “Knuckles” from the *Sonic* video games |

Note: These are obviously not real emails, but these are simple emails that are used to test the addition of new users. Also, all user-entered data entered into a database are sanitized using both htmlspecialchars() and real\_escape\_string() when being inserted. That way, they can’t be recognized as HTML or SQL when being retrieved from the database, preventing HTML/SQL injections. Passwords are salted before insertion. You cannot create an employee with the same email as a customer, and employees have “priority” over customers: when logging in, the system will search for matching employee emails first, then matching customer emails.

**TYPES OF TRANSACTIONS SUPPORTED BY USERS:**

1. Insert entries into 'reservations' table (as customer)
2. Update entries in 'customers' table (as customer)
3. Delete entries from 'reservations' table (as customer)
4. Insert (hire) new employees (as manager)
5. Delete (fire/dismiss) existing employees other than themselves (as manager)
6. Update employee information (as manager)
7. Update room type information (as manager)
8. Update room price information (as manager)

Note: Both customers and managers/administrators are able to update, insert, or delete some data in the database

**WEBSITE/DATABASE IMPLEMENTATION NOTES AND COMMENTS:**

1. The "employee" view uses a navbar with white background, and the "customer" view uses a navbar with a black background. This helps to distinguish both website views from each other.
2. If the price of a room changes, existing customers with bookings will be charged the price at booking time. That means poor Sonic Hedgehog has to pay $409/night for his reservation of Room 106 from Sept 2 to 9, and won't be able to take advantage of my later price drop afterward to $389 unless he cancels and rebooks his reservation. Conversely, if the price goes up later, you’re locked in at the lower price unless you delete your reservation.
3. Reservations can only be deleted up to the day before the reservation starts.
4. My prices are a tad expensive, but these are luxury rooms after all (not like those run-down hotels on Huron Church Road), and my hotel has only 6 rooms.
5. I used jQuery version 1.12.4 (not the current one) since the Bootstrap "Get Started" page used it when I started to work on the project. They've changed it in the past few days to 3.2.1, however. I downloaded jQuery 1.12.4 from here: <http://minime.stephan-brumme.com/jquery/1.12.4/>
6. The font used is “Fira Sans”, from here: <https://fonts.google.com/specimen/Fira+Sans>. Unfortunately, this has to be loaded from Google Fonts on each page because they have this aversion to allowing you to download their resources.
7. Bootstrap is used to make the website responsive. It renders the website differently for devices at a breakpoint of 768px. I also use a media query to make some content render differently (usually tables and large fonts) for devices under 500px.
8. The home page footer’s “X vacancies available tonight” reports the number of 1-night vacancies available with a start date of today (where today is calculated using PHP).
9. The database information used to log into the database is located in a file called “databaseinfo.php”, along with the statement that creates the MySQLI $conn variable on each page. This file is not included in the GitHub repository for security reasons.

**DATABASE ACCESS TECHNIQUES USED:**

1. Search rooms by availability from [DATE] to [DATE + N days], where 1 <= N <= 7
2. Search rooms by number of beds in the room (1 or 2 or any)
3. Order rooms by price
4. Order room by room number (descending or ascending)
5. Occupancy rate by room number
6. Occupancy rate for total hotel
7. Hotel revenue by week for whole hotel
8. Hotel revenue by week by room number

**HOW TO TEST ASPECTS OF THE WEBSITE (CUSTOMER):**

1. To create a new user, access “signup.php” through the “Sign Up” link on the navbar.
2. To login, either create a new user, or log in as an existing user via “login.php”, through the “Log In” link in the navbar.
3. To browse the rooms available, click on “Rooms” in the navbar. If you’re not logged in, you’ll be forced to log in before proceeding. The login page automatically redirects you to the page you were visiting before you logged in, with “index.php” as a fallback.
4. Once selecting the filters on the Rooms page, the content will be replaced by rooms that match the search parameters. Click on “Confirm Reservations” to confirm your reservations and pay for them.
5. You are redirected to a page to pay for your reservation. I don’t actually store your credit card information; in real life, that’s extremely unsafe and contracted to other companies (source: <https://stackoverflow.com/questions/3328922/saving-credit-card-information-in-mysql-database>). I also wouldn’t know how to “un-encrypt” the credit card to issue refunds for cancelled reservations, either. Just pretend the credit card information is accepted in the background. However, you will get a warning if you try to pay with an expired credit card, though.
6. When confirmed, you will receive a message, along with a link to a page (Account and Reservations) where you can view your booked reservations, delete reservations for future dates, and change your account info.

**HOW TO TEST ASPECTS OF THE WEBSITE (MANAGER):**

1. Log in using the same “Log In” link that customers use.
2. You will be redirected to the Management page, which has forms to change room descriptions, room prices, add new employees, and delete existing employees (other than yourself, of course). Other pages are accessed through the navbar.
3. The “Update Account” page allows you to change your own personal information. You can’t change the information of other managerial users once it’s created – they have to do that themselves using their accounts.
4. The “Reports” page gives a weekly summary of room occupancy (number of nights each room is rented) and revenue, for the previous, current, and next week.