**Introduction**

The project database platform is mysql Ver 8.0.25-0ubuntu0.20.04.1 for Linux on x86\_64 ((Ubuntu)). It will be the primary source of all user-related data and run on a Digital Ocean hosted server along with the API. The database will possess several stored procedures meant for adding, editing, and removing data from the appropriate tables, as well as returning results and data from said tables.

To access the database, first connect to the Digital Ocean server through PuTTy or another capable software. Next, run the command ‘mysql -u insert\_database\_username\_here -p’. Upon entering your password, you will be given access to the database. Before continuing, run the command ‘use COP4331’ to set the target database to the project’s.

**Tables**

* **Users**: The primary database table for storing user information. It has 7 columns named ID, DateCreated, DateLastLoggedIn, Firstname, LastName, Username, and Password. ID keeps track of the unique User ID for every user of the site in the form of an INT. DateCreated keeps track of the date and time that the user account was created in the form of a DATETIME. DateLastLoggedIn tracks the most recent date and time a user logged in to the site and is represented as a DATETIME as well. FirstName and LastName are the first and last names of the suer respectively and are both represented as VARCHAR(50). Username and Password are also VARCHAR(50)s and store the username and the password of each user respectively.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | DateCreated | DateLastLoggedIn | FirstName | LastName | Username | Password |
| 1 | 5/21/2021 3:47 | 5/21/2021 20:21 | Rick | Leinecker | RickL | COP4331 |
| 2 | 5/21/2021 6:30 | 5/22/2021 7:40 | Fake | Person | TestAccount | 12345 |
| … |  |  |  |  |  |  |

* Contacts: The primary database table for storing user contact information. It has 7 columns named ID, DateCreated, FirstName, LastName, Phone, Email, and ContactID. ID is an INT reperesenting a unique contact identifier for each contact added. DateCreated is a DATETIME keeping track of when the contact was added. FirstName and LastName are VARCHAR(50)s that store the first and last name of the contact respectively. Phone is a VARCHAR(50) that stores the phone number of the contact. Email is a VARCHAR(50) that stores the mail address of the contact. ContactID is an INT that keeps track of which User ID is associated with the contact.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | DateCreated | FirstName | LastName | Phone | Email | ContactID |
| 1 | 5/21/2021 3:47 | Pizza | Hut | 4073595699 | [pizzathehut@gmail.com](mailto:pizzathehut@gmail.com) | 2 |
| 2 | 5/21/2021 6:30 | Dwayne | Johnson | 4074074077 | [therock@yahoo.com](mailto:therock@yahoo.com) | 1 |
| … |  |  |  |  |  |  |

In this example, Pizza Hut is a contact of Fake Person and Dwayne Johnson is a contact of Rick Leinecker.

**Stored Procedures**

* Login: Will search through the Users table and return a User ID if it finds a row with a matching Username and Password pair to the passed in parameters. Will return NULL otherwise. [Parameters are usr <VARCHAR(50)> and pwd <VARCHAR(50)>]
* Register: Will search the Users table to make sure that there is no Username that matches the one passed in with the parameters. If there are none, it will add the user and return the User ID. Otherwise it will return NULL. [Parameters are fname <VARCHAR(50)>, lname <VARCHAR(50)>, usr <VARCHAR(50)>, and pwd <VARCHAR(50)>]
* DeleteUser: Used to delete a user from the Users table based on matching ID. [Parameter is userid <INT>]
* AddContact: Will make sure there is not already a contact with the same matching Phone, Email, and ContactID as the input parameters. If not, it will add a contact to the Contacts table and return the ID and otherwise return NULL. [Parameters are fname <VARCHAR(50)>, lname <VARCHAR(50)>, phonenum <VARCHAR(50)>, emailadd <VARCHAR(50)>, userid <INT>]
* DeleteContact: Will remove a contact with the ID from the Contacts table. [identifier <INT>]
* EditContact: Will overwrite the Contacts table row with a new first name, last name, phone number, and email with the matching ID. [Parameters are fname <VARCHAR(50)>, lname <VARCHAR(50)>, phonenum <VARCHAR(50)>, emailadd <VARCHAR(50)>, identifier <INT>]
* SearchContacts: Will return all contact data for every row in the Contacts table that contains the first and last name search query and has a ContactID that matches the userid. If no first name or last name is supplied, then the stored procedure will return all contacts associated with the userid. [Parameters are fname <VARCHAR(50)>, lname <VARCHAR(50)>, userid <INT>]