

# **Pi In The Sky**

## **Milestone #3**

### **Database Management System**

#### **The Architecture**

We are using MariaDB as our database tool. Currently, our database is designed with two tables. The first table, `users`, consists of user login information. It has the user's name, their unique login username and encrypted password, their email, a unique key. The Pi comes with default login credentials and the user can then change the default password and later add more users. The second table, `websites`, is log of all websites that have created (uploaded) by all users. It stores the username of the user who created the website, when the website was created, the name of the website, whether the website is a registered domain name, whether the website is currently active and finally, where the website source code is stored on the server. A normal user would be able to view only the websites they created. We are currently in the process of creating a superuser who can view the list of all websites created by all users.

As the cloud storage feature of our project matures and becomes more wholesome, there will be another table consisting of information about all files that are uploaded such as their uploaded date, when it was last accessed, the size of the file, sharing information (users who have access to it), etc. But, our current database architecture consists of only the two tables described above.

*Figure 1* shows the relationship between the two tables – `users` and `websites`. As you can see, it maps a one-to-many function i.e., a single user can create multiple websites, but a website can have only one owner (can be created by only one person).

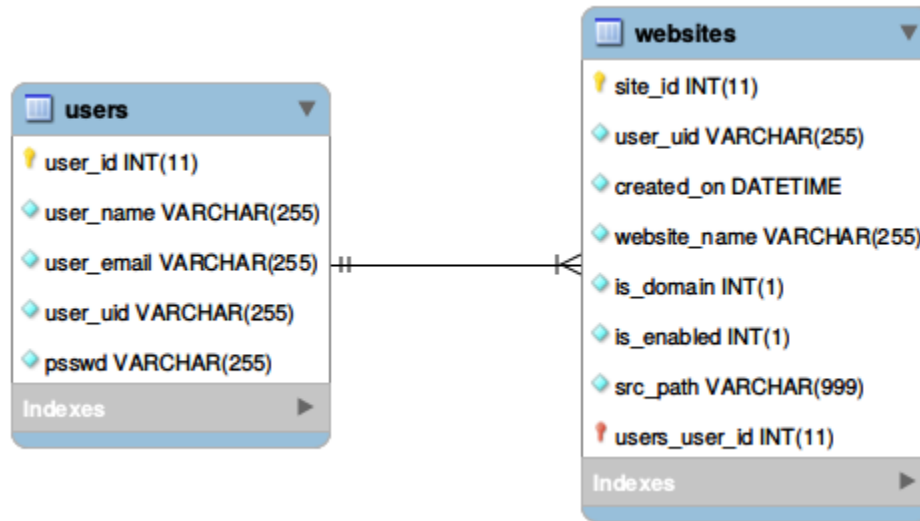


Figure 1 Entity Relationship diagram for our current database architecture

## Populating the Database

As mentioned earlier, the **users** table initially comes with a single login credential. Using this the user will be able to create multiple other users which populates the table on the server. This is done with PHP via MySQL queries. A user will also be able to change their password.

The **websites** table, on the other hand is initially unpopulated. Whenever a user successfully uploads a website, the PHP script calls MySQL queries that populate the table at the backend.

## Scripts

The scripts that create and initialize the database can be found inside the **mysqlScripts** folder in root of the GitHub repository. Specifically:

- **createDB.sql** – Creates the database  
Link: <https://github.com/dandrewbarlow/Pi-in-the-Sky/blob/master/mysqlScripts/createDB.sql>
- **initDB.sql** – Populates the database with initial values  
Link: <https://github.com/dandrewbarlow/Pi-in-the-Sky/blob/master/mysqlScripts/initDB.sql>