Quack Time Installation Guide

Authors: Neal Kimchi, Nikhar Ramlakhan, Areyan Rastawan, Abie Safdie

Table of Contents

Installing Quack Time

This section provides a comprehensive guide for setting up and getting started with Quack Time.

STEP 1: Install Python

First, you need to have Python 3.12 installed on your machine.

1. Download the Python 3.12 installer from the official Python website: https://www.python.org/downloads/
2. Run the installer and follow the on-screen instructions.
3. Verify your installation by opening your terminal (macOS/Linux) or Command Prompt (Windows) and type:

```
python3 –-version # For macOS / Linux
python --version  # For Windows
```

You should see `Python 3.12.x` in the output.

STEP 2: Install Packages

Next, you need to install the required dependencies for Quack-Time.

macOS / Linux

Open your terminal and run the following commands:

```
# Ensure pip is up-to-date
python3 -m pip install --upgrade pip

# Install required packages
pip3 install bcrypt
pip3 install Flask
pip3 install mysql-connector-python
pip3 install --upgrade mysql-connector-python
```

Windows

Open Command Prompt and run the following commands:

```
# Ensure pip is up-to-date
python -m pip install --upgrade pip

# Install required packages
pip install bcrypt
pip install Flask
pip install mysql-connector-python
```

STEP 3: Running Quack Time

Now that you have Python and the necessary packages installed, you can run the Quack-Time web app.

macOS / Linux

1. Open your terminal.

2. Navigate to the directory containing `app.py`. For example:

```
cd path/to/Quack Time
```

3. Start the application:

```
python3 app.py
```

4. Open your web browser and go to `http://localhost:5002`

Windows

1. Open your Command Prompt.

2. Navigate to the directory containing `app.py`. For example:

```
cd path/to/Quack Time
```

3. Start the application:

```
python app.py
```

4. Open your web browser and go to `http://localhost:5002`

From an IDE

You can also run Quack-Time from any Integrated Development Environment (IDE) of your choice. Refer to your specific IDE's documentation on how to run a Python script.

Server Installation Setup Guide

Note: Server Installation Not Required

These instructions are intended for users who wish to set up a private database alongside their Quack Time application. By default, the source code connects to the existing master Quack Time server. Hence, you can skip this server installation process.

This section provides a comprehensive guide for setting up a separate MySQL server on a Linux-enabled machine via the Linux terminal. The objectives of this guide are to:

- Install a MySQL server on a Linux enabled machine such as ix-dev
- Create an account that allows remote access to the database.
- Change the database configuration file in the source code.

STEP 1: Log on to the Linux-enabled server

To begin, log on to a Linux-enabled machine, such as the University of Oregon Computer Science department machine ix, using the following terminal command:

```
ssh username@ix.cs.uoregon.edu
```

Replace “username” with your login name if you are using the ix-dev server.

STEP 2: Install MySQL

Navigate to your home directory and run the following command to install MySQL:

```
mysqlctl install
```

During installation, you'll be prompted to enter a password of your choice. Remember this password for future MySQL commands.

STEP 3: Start the MySQL Server

Initiate the MySQL server by running:

```
mysqlctl start
```

This command not only starts the server but also creates a .my.cnf file and generates a port.

STEP 4: Verify server status

Check the status of the MySQL server by running:

```
mysqlctl status
```

Make note of the port number displayed in the output. It will be used to connect to the server remotely. An example output is*: mysqld (pid 898545) listening on ix-dev:3932* where the port is 3932.

Alternatively, if you have administrative account privileges on your Linux machine, use the command:

```
mysqladmin -p version
```

And note the TCP port number.

STEP 5: Access the MySQL terminal

Enter the MySQL query terminal by running:

```
mysql -p
```

You'll see mysql> on the left side, indicating that you can now enter queries.

STEP 6: Create a database

Create the Active Reading Assistant database by typing:

```
CREATE DATABASE quack_time;
```

STEP 7: Create a user and grant privileges

Create a user with the following query:

```
CREATE USER 'username'@'%' IDENTIFIED BY 'password';
```

Replace "username" and "password" with your desired credentials. Keep the ' marks.

Grant privileges to the database with the following query:

```
GRANT ALL PRIVILEGES ON quack_time.* TO 'username'@'%' WITH GRANT OPTION;
```

Replace "username" with the username identified when creating a user. Keep the ' marks.

STEP 8: Create database tables

Run the following queries to create necessary tables:

```
CREATE TABLE quack_time.Users ( DuckID CHAR(9) UNIQUE PRIMARY KEY, FirstName VARCHAR(255), LastName VARCHAR(255), PassHash VARCHAR(255), PassSalt VARCHAR(255), CONSTRAINT CHK_DuckID_Length CHECK (LENGTH(DuckID) = 9) );
```

```
CREATE TABLE quack_time.TaskBoard ( BoardID INT AUTO_INCREMENT PRIMARY KEY, DuckID CHAR(9) NOT NULL, BoardName VARCHAR(255), FOREIGN KEY (DuckID) REFERENCES Users(DuckID) );
```

```
CREATE TABLE quack_time.Tasks ( TaskID INT AUTO_INCREMENT PRIMARY KEY, BoardID INT, TaskName VARCHAR(255), TimeRemaining INT, FOREIGN KEY (BoardID) REFERENCES TaskBoard(BoardID) );
```

```
CREATE TABLE quack_time.Logs ( LogID INT AUTO_INCREMENT PRIMARY KEY, DuckID CHAR(9) NOT NULL, LogDate DATETIME, BoardName VARCHAR(255), TaskName VARCHAR(255), TimeSpent INT, Rating INT CHECK (Rating BETWEEN 1 AND 5), FOREIGN KEY (DuckID) REFERENCES Users(DuckID) );
```

STEP 9: Finalize setup

Exit the MySQl query terminal by typing:

```
exit
```

Your server is now set up and the user can exit the terminal accessing the Linux server.

STEP 10: Configure the Server to Run with the Web App

1. Locate the `db.py` file:
   - Navigate to the `Quack Time` source folder where the `db.py` file is located.
2. Open the `db.py` file:
   - Open the `db.py` file in a text editor or IDE of your choice (e.g., Notepad, Visual Studio Code, PyCharm).
3. Update the Database Connection Configuration.
   - In the `db.py` file, navigate to line 17 and update the following code block with your database connection details:

```
# Database connection configuration
db_config = {
    'host': 'ix-dev.cs.uoregon.edu',
    'port': your_port_number, # Replace with the port number noted earlier (step 4)
    'user': 'your_username',  # Replace with your MySQL username (step 7)
    'password': 'your_password', # Replace with your MySQL password (step 7)
    'database': 'quack_time'
}
```

Replace `your_port_number`, `your_username`, and `your_password` with your specific details. Save the file after making these changes.