

411 Final Project Documentation

Group Members: Andrew Xin, Tong Zhang, Sarah Lam, Jeremy Lau

Emails: andrewx@bu.edu, alicoooo@bu.edu, sarahl@bu.edu, jlau1@bu.edu

Project Repository: <https://github.com/andrewx-bu/WeatherDashboard>

Note - sandbox branch: <https://github.com/andrewx-bu/WeatherDashboard/tree/sandbox>

411 Final Project Documentation.....	1
1. Steps to run our application:.....	1
2. Variables defined in our env file:.....	3
3. Smoketest output:.....	4

1. Steps to run our application:

Windows:

- I. Download/ clone repo
 - A. git clone <https://github.com/andrewx-bu/WeatherDashboard.git>
- II. Using terminal in vscode, code editor of choice, or in command prompt:
 - A. cd into the project directory
 - B. python setup_db.py
 - C. python app.py
- III. Using a separate command prompt,:
 - A. cd into the project directory
 - B. Run curl commands (examples given below):

Windows example for our routes:

- curl -X POST http://127.0.0.1:5000/create-account -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'
- curl -X GET http://127.0.0.1:5000/login -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'
- curl -X PUT http://127.0.0.1:5000/update-password -H "Content-Type: application/json" -d '{"username": "user1", "old_password": "password1", "new_password": "password2"}'
- curl -X POST http://127.0.0.1:5000/api/add-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "location": "San Francisco"}'

Example of running routes:

```
C:\Windows\System32\cmd.exe x + v
C:\Users\jerem\OneDrive\Documents\A_BU\Senior\411\finalproj411>curl -X POST http://127.0.0.1:5000/create-account -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'
{
  "message": "Account created",
  "status": "success"
}

C:\Users\jerem\OneDrive\Documents\A_BU\Senior\411\finalproj411>curl -X GET http://127.0.0.1:5000/login -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'
{
  "message": "Login successful",
  "status": "success"
}

C:\Users\jerem\OneDrive\Documents\A_BU\Senior\411\finalproj411>curl -X PUT http://127.0.0.1:5000/update-password -H "Content-Type: application/json" -d '{"username": "user1", "old_password": "password1", "new_password": "password2"}'
{
  "message": "Password updated",
  "status": "success"
}

C:\Users\jerem\OneDrive\Documents\A_BU\Senior\411\finalproj411>curl -X POST http://127.0.0.1:5000/api/add-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "location": "San Francisco"}'
{
  "message": "Favorite location added",
  "status": "success"
}
```

Check db for favorites here: http://127.0.0.1:5000/api/get-favorites?user_id=1

```
← → ↻ ⓘ 127.0.0.1:5000/api/get-favorites?user_id=1
Pretty-print ☐
{
  "favorites": [
    {
      "created_at": "2024-12-10 19:47:13",
      "id": 1,
      "location": "San Francisco",
      "updated_at": "2024-12-10 19:47:13"
    }
  ]
}
```

Check db for users here: <http://127.0.0.1:5000/get-all-users>

```
← → ↻ ⓘ 127.0.0.1:5000/get-all-users
Pretty-print ☐
{
  "users": [
    {
      "id": 1,
      "username": "user1"
    }
  ]
}
```

Routes for salted passwords / user accounts: (Mac)

1. **create_account**
 - a. `curl -X POST http://127.0.0.1:5000/create-account -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'`
 - b. `curl -X POST http://127.0.0.1:5000/create-account -H "Content-Type: application/json" -d '{"username": "user2", "password": "password2"}'`
2. **delete_account**
 - a. `curl -X DELETE http://127.0.0.1:5000/delete-account -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'`
3. **login**
 - a. `curl -X GET http://127.0.0.1:5000/login -H "Content-Type: application/json" -d '{"username": "user1", "password": "password1"}'`
4. **update_password**
 - a. `curl -X PUT http://127.0.0.1:5000/update-password -H "Content-Type: application/json" -d '{"username": "user1", "old_password": "password1", "new_password": "password2"}'`
5. **get_all_users**
 - a. <http://127.0.0.1:5000/get-all-users>

Routes for weather feature (Mac)

1. **add_favorite:** `curl -X POST http://127.0.0.1:5000/api/add-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "location": "San Francisco"}'`
 - a. Run these in another bash terminal; [blue](#) is the JSON being passed in
2. **remove_favorite:** `curl -X DELETE http://127.0.0.1:5000/api/remove-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "location": "San Jose"}'`
3. **update_favorite:** `curl -X PUT http://127.0.0.1:5000/api/update-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "old_location": "San Francisco", "new_location": "San Jose"}'`
4. **clear_favorites:** `curl -X DELETE http://127.0.0.1:5000/api/clear-favorites -H "Content-Type: application/json" -d '{"user_id": "1"}'`
5. **get_favorites:** http://127.0.0.1:5000/api/get-favorites?user_id=1
 - a. `curl -X POST http://127.0.0.1:5000/api/add-favorite -H "Content-Type: application/json" -d '{"user_id": 1, "location": "San Francisco"}'`

2. Variables defined in our env file:

We removed our secrets and put them in a text file called “env.txt” in our repo that contains:

```
OPENWEATHER_API_KEY=[INSERT YOUR KEY HERE]
DB_PATH=/app/db/song_catalog.db
SQL_CREATE_TABLE_PATH=/app/sql/create_song_table.sql
CREATE_DB=true
```

3. Smoketest output:

- Database set up, app is running (Mac)

```
● (.venv) crc-dot1x-nat-10-239-2-85:WeatherDashboard andrew$ sh smoketest.sh
Starting smoke tests...
Checking health status...
Service is healthy.
Checking database connection...
Database connection is healthy.
Creating user: testuser0
User created successfully.
Logging in user: testuser0
Login successful.
Updating password for user: testuser0
Password updated successfully.
Retrieving all users...
Users retrieved successfully.
Fetching user ID for username: testuser0
User ID for testuser0 is 1.
Adding favorite location: New York for user: 1
Favorite location added successfully.
Adding favorite location: San Francisco for user: 1
Favorite location added successfully.
Fetching favorite locations for user: 1
Favorites retrieved successfully.
Updating favorite location from 'New York' to 'Boston' for user '1'...
Favorite location updated successfully.
Fetching favorite locations for user: 1
Favorites retrieved successfully.
Removing favorite location 'San Francisco' for user '1'...
Favorite location removed successfully.
Fetching favorite locations for user: 1
Favorites retrieved successfully.
Clearing all favorites for user '1'...
All favorites cleared successfully.
Deleting user: testuser0
User deleted successfully.
Testing coordinates retrieval for city: New York
Successfully fetched coordinates for city: New York
Testing weather forecast for city: New York
Successfully fetched weather forecast for city: New York
Testing air pollution forecast for city: New York
Successfully fetched air pollution forecast for city: New York
Testing current weather for city: New York
Successfully fetched current weather for city: New York
Testing air pollution data for city: New York
Successfully fetched air pollution data for city: New York
Testing coordinates retrieval for city: Los Angeles
Successfully fetched coordinates for city: Los Angeles
Testing weather forecast for city: Los Angeles
Successfully fetched weather forecast for city: Los Angeles
Testing air pollution forecast for city: Los Angeles
Successfully fetched air pollution forecast for city: Los Angeles
Testing current weather for city: Los Angeles
Successfully fetched current weather for city: Los Angeles
Testing air pollution data for city: Los Angeles
Successfully fetched air pollution data for city: Los Angeles
All smoke tests completed successfully!
○ (.venv) crc-dot1x-nat-10-239-2-85:WeatherDashboard andrew$ █
```