CS 474 Capstone Proseminar - Individual Project

Natalie Lee
nml3@hood.edu

Fall 2023

Question 1

This project will be a Weather Condition Querying Web Application deployed on a Raspberry Pi. The app will provide users with information on current and historical weather conditions for specific locations. This web application will be implemented with the Flask Python framework for back-end/front-end development, database management (using PyMongo), and API handling. This app will utilize the free OpenWeatherMap API.

Functions:

- **Location Management:** Users can add new locations, view existing locations, update location details, and delete saved locations.
- **Weather Data Retrieval:** Users can retrieve weather conditions for specified locations and dates.
- **Weather Data Visualization:** Users can view graphs and plots to compare current and historical weather condition data.

Technologies:

- **OpenWeatherMap API:** The external API utilized for retrieving weather condition data. `https://openweathermap.org/api`
- **Raspberry Pi and Touchscreen Display:** The Raspberry Pi 3 Model B, a single-board computer with wireless LAN and Bluetooth connectivity will be used to deploy this app. The Pi Foundation Display, a 7" Touchscreen Display for Raspberry Pi will provide the user interface for interacting with the Raspy Weather application.
- **Git and Github:** Utilized for version control.

Question 2

Resources:

1. `https://www.youtube.com/watch?v=JCD7YdOSsWI`

   Description: A YouTube video describing how to build a simple weather querying Flask app specifying a city using the OpenWeatherMap API. I will use this as a baseline for how to connect a Flask app to the API I will be using.

   This video does not include a database and historic data querying and visualization.

2. `https://www.youtube.com/watch?v=E2hytuQvLlE`

   Description: A YouTube video describing how to add graphs to a Flask App. I will use this for historic weather data visualization.

   This video does not include how to specifically get data from an API or format it for weather data.

3. `https://www.youtube.com/watch?v=kZlet-0epxE`

   Description: A YouTube video describing the basics of using MongoDB in a Flask App with CRUD functionality.

   This video does not include how to specifically use these tools for my web app goals.

Question 3

Week	Tasks to be Completed
1	• Update Raspberry Pi OS • Mock-up website wireframes • Create Flask App Boilerplate • Create UI for location management and UI forms for weather data querying • Integrate OpenWeatherMap API to Flask App.
2	• Implement weather data visualization graphs in Flask App • Implement MongoDB CRUD functionality for managing historical weather data associated with specific locations
3	• Work on user interface improvements • Make a case for the Raspberry Pi and Touch Screen Display
4	• Deploy the web app on Raspberry Pi • Perform testing and debugging • Create slide deck for project presentation

Table 1: Project Task Breakdown by Week

Question 4

GitHub Repository: `https://github.com/natalieswork/RaspyWeather`