**1. Introduction**

This project focuses on developing a system that retrieves real-time weather data for various locations using OpenAI's API. The motivation behind this project is to create an easy-to-use interface for users to obtain weather information seamlessly.

**2. Objectives**

Outline the main objectives of the project.

**Example:**

* To create a system that retrieves weather data based on user input for latitude and longitude.
* To utilize OpenAI's API for generating conversational responses regarding weather.
* To build a user-friendly interface for interacting with the weather data.

**3. Design**

**3.1 Architecture**

The system consists of a client-side interface where users input location data (latitude and longitude). The backend processes the data and retrieves weather information from a CSV file, providing responses through OpenAI’s API.

**3.2 Technologies Used**

List the technologies and programming languages utilized in the project.

**Example:**

* Python
* OpenAI API
* Pandas (for data manipulation)
* JSON (for data interchange)

**4. Implementation**

**4.1 Code Explanation**

The main functionality of the project is encapsulated in functions that handle:

* Loading weather data from a CSV file.
* Fetching current weather information based user prompt, fetches latitude and longitude from csv data.
* The retrieves weather information respective coordinates and displays to user.

**4.2 Functionality**

Users input their desired location's name. function call fetches latitude and longitude and then system checks the CSV data for matching coordinates and retrieves the corresponding weather information, which is then formatted and returned to the user.

**5. Challenges**

One significant challenge was sourcing the data.

**6. Conclusion**

The weather data retrieval system successfully meets its objectives by providing accurate weather information to users based on their location. Future enhancements could include the integration of a user interface and real-time weather updates from an external API.