COMPUTER ENGINEERING WORKSHOP

S.E. (CIS) OEL REPORT

Project Group ID:

Muhammad Ahad Khan	CS-23049
Muhammad Umar	CS-23072
Shayan Malik	CS-23053

BATCH: 2023

Department of Computer and Information Systems Engineering

NED University of Engg. & Tech.,
Karachi-75270

CONTENTS

S.No.

1. **Problem Description**
2. **Methodology**
3. **Results**

PROBLEM DESCRIPTION

Construct an integrated environmental monitoring system in C, covering a range of fundamental concepts and practical applications. The project involves interacting with a free API that provides real-time environmental data. The system's core functionalities include data retrieval, processing and reporting

METHODOLOGY

`api_handler.c` and `api_handler.h`:

- **Purpose**: Fetches raw weather data from the Open Weather API using the city name and API key.
- **Method**:
  - A **`CURL`** HTTP GET request fetches the data.
  - The response is dynamically stored in memory using a **`writeCallback`** function.
  - The raw data is also saved to a file (**`raw_data.json`**) for debugging or archival purposes.
- **Key Functionality**:
  - **`fetch_weather_data`** retrieves JSON-formatted weather data(**api_handler**).

`json_parser.c` and `json_parser.h`:

- **Purpose**: Parses the raw JSON response to extract and structure relevant weather information.
- **Method**:
  - Uses the **`json-c`** library to process fields like temperature, humidity, description, wind speed, etc.
  - The parsed data is stored in a **`WeatherData`** structure.
  - Includes a notification mechanism if temperature exceeds 15°C **(uses `notify-send`)**.
- **Key Functionality**:
  - **`parse`** parses raw data and validates its correctness(**json_parser**).

`Processor.c` and `processor.h`:

- **Purpose**: Converts parsed weather data into a readable format and writes it to a file (**`weather_data.txt`**).

- **Method**:
  - Processes numerical fields (e.g., converting temperature from Kelvin to Celsius).
  - Writes formatted output to a text file for user access.
- **Key Functionality**:
  - **`write_data_into_file`** saves weather details to a file(processor)

`main.c:`

- **Purpose**: The main driver of the program.
- **Method**:
  - Fetches weather data for a specified city (**`Karachi`**) using an API key.
  - Parses the fetched data.
  - Processes and saves the data if valid; displays error messages otherwise.
- **Key Functionality**:
  - Controls the flow of execution, integrates API handler, parser, and processor modules(**main**).

`raw_data.json`

- **Purpose**: Stores raw weather data fetched from the OpenWeather API.


```
{
"coord": {"lon": 67.0822, "lat": 24.9056},
"weather": [{"main": "Clouds", "description": "few clouds"}],
"main": {"temp": 300.05, "feels_like": 300.91, "humidity": 57},
"wind": {"speed": 2.57, "deg": 260},
"name": "Karachi"
}
```


`shell_scripting.sh:`

- **Purpose**: Manages output files for long-term storage and archival.
- **Method**:
  - Maintains a directory (**`./files`**) to store up to 7 recent weather reports.
  - Deletes the oldest file when the limit is exceeded.
  - Appends weather data from **`weather_data.txt`** into a timestamped file within this directory.
- **Key Functionality**:
  - Automates file organization and ensures clean storage(**shell_scripting**).

Process Flow

1. **Fetch Weather Data**:
   - Input: City name (`Karachi`) and API key.
   - Output: Raw JSON response saved in `raw_data.json`.
2. **Parse Weather Data**:
   - Input: Raw JSON.
   - Output: Structured data in `WeatherData` format.
3. **Save Processed Data**:
   - Input: Structured data (`WeatherData`).
   - Output: Human-readable weather report saved in `weather_data.txt`.
4. **File Management**:
   - Automates storage of weather reports in a dedicated directory.

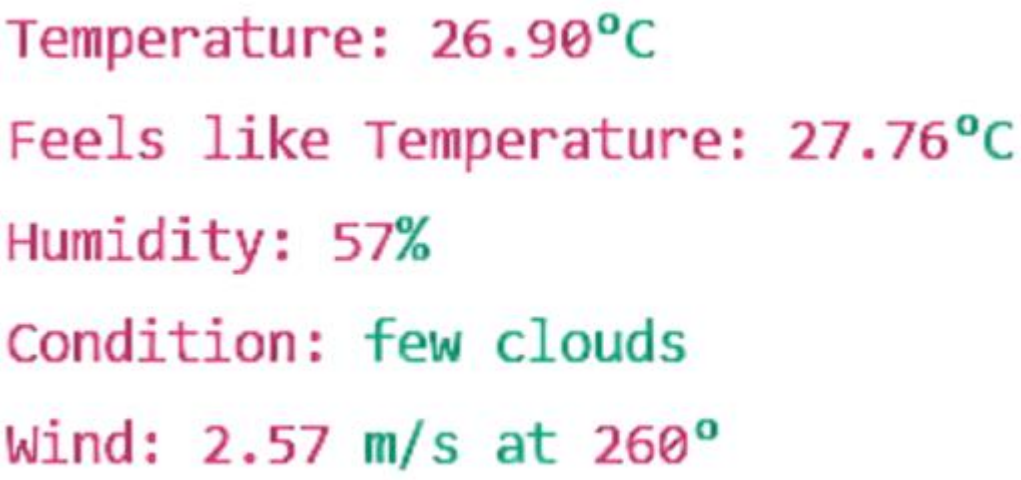
RESULTS

Raw Weather Data (`raw_data.json`)

This file contains the raw JSON response fetched from the OpenWeather API for the city "Karachi.

Formatted Weather Report (`weather_data.txt`)

The program processed the JSON data and generated the following weather report for Karachi:



DEPARTMENT OF COMPUTER & INFORMATION SYSTEMS
ENGINEERING BACHELORS IN COMPUTER SYSTEMS ENGINEERING
Course Code: CS-219
Course Title: Computer Engineering Workshop
Open Ended Lab
SE Batch 2023, Fall Semester 2024
Grading Rubric
TERM PROJECT

Group Members:

Student No.	Name	Roll No.
S1	**Muhammad Ahad Khan**	**CS-23049**
S2	**Muhammad Umar**	**CS-23072**
S3	**Shayan Malik**	**CS-23053**

CRITERIA AND SCALES				**Marks Obtained**		
				S1	**S2**	**S3**
Criterion1: Has the student implemented an efficient and scalable solution for data retrieval, processing, and reporting?						
0	1	2	3			
The student has not even implemented a basic solution that meets the project's requirements.	The student has implemented a basic solution that meets the project's requirements but may lack optimization in certain aspects.	The student has implemented a proficient and well-optimized solution.	The student has implemented an exceptionally efficient and scalable solution.			

Criterion 2: Has student demonstrated a strong understanding of C programming fundamentals?						
0	1	2	3			
The student doesn't have basic understanding of C programming fundamentals.	The student exhibits a basic understanding of C programming fundamentals.	The student demonstrates a strong understanding of C programming fundamentals.	The student demonstrates an exceptional understanding of C programming fundamentals.			
Criterion 3: How well written is the report?						
0	1	2	3			
The submitted report is unfit to be graded.	The report is partially acceptable.	The report is complete and concise.	The report is exceptionally written.			
			Total Marks:			