

**Computer Engineering Workshop**

**OPEN ENDED LAB**

CS 22141: Yahya Alyan

CS 22137: Farzam Nasir

CS 22128: Ahmed Khan

**CHAPTER 1: 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, analysis, and reporting.

* DataRetrieval**:** 
  + Interact with a free API to retrieve real-time environmental data (e.g., temperature, humidity).

* Data Processing and Analysis:
  + Process the retrieved data using C programming concepts.
  + Implement algorithms to analyze environmental data for anomalies or trends.

* Data Storage:
  + Store both raw and processed environmental data in files.

* Reporting System:
  + Develop a reporting system that generates reports based on the analyzed environmental data.

* Automation and Integration:
  + Create shell scripts to automate tasks such as data retrieval, processing, and report generation.
  + Integrate shell scripts with the C program to enhance automation.

* Optimization and Efficiency:
  + Utilize pointers and dynamic memory allocation in the C program to optimize data manipulation and enhance efficiency.

* Real-time Alerts:
  + Implement real-time alerts using Linux system calls to notify relevant personnel of critical environmental readings.

* Documentation and Code Organization:
  + Develop comprehensive documentation for the monitoring system.
  + Use header files to modularize the C code and enhance code readability.

# Chapter 2: Methodology

* **1.1: Overview:**

The implemented weather monitoring system is designed to fetch real-time weather data using the Open Meteo API, process the received data, and generate a report. The system focuses on temperature, humidity, wind speed, wind direction, surface pressure, and sea-level pressure.

* **1.2: Implementation:**

Hello.

**1.2.1: Data Retrieval:**

The system uses the “libcurl” library to make an HTTP request to the Open Meteo API. The requested data includes the current temperature, humidity, wind speed, wind direction, surface pressure, and sea-level pressure.

**1.2.2: Data Processing:**

After receiving the raw JSON data, the system parses it using the jansson library. It then extracts the relevant weather parameters and stores them in the "processed\_data.json" file for further analysis.

**1.2.3: Threshold Alerts:**

The system checks for anomalies by comparing the obtained values with predefined thresholds. If any parameter exceeds its threshold, an alert is displayed, and notifications are sent. Currently, the system handles temperature and wind speed alerts.

**1.2.4: Time-Based Handling:**

The system has a time-based handling mechanism implemented through the “Handling()” function. It checks if the current day is Monday or Sunday and takes corresponding actions, including generating weekly reports and clearing temporary files.

**1.2.5: Report Generation:**

The processed data is formatted and printed on the console. Additionally, a summary report is created in the "processed\_data.json" file. The system also generates a weekly report on Sundays, containing the mean temperature for the week.

# Chapter 2: Results

* **2.1: Raw Data Storage:**

The raw data retrieved from the Open Meteo API is stored in the "raw\_data.json" file, providing a record of the API responses.

* **2.2: Processed Data:**

The processed weather data, including temperature, humidity, wind speed, wind direction, surface pressure, and sea-level pressure, is stored in the "processed\_data.json" file.

* **2.3: Alerts:**

The system generates alerts and notifications for high temperatures and wind speeds, providing timely information to the user.

* **2.4: Weekly Reports:**

On Sundays, the system calculates the mean temperature for the week and generates a weekly report in the "Weekly\_Weather\_Report.txt" file.

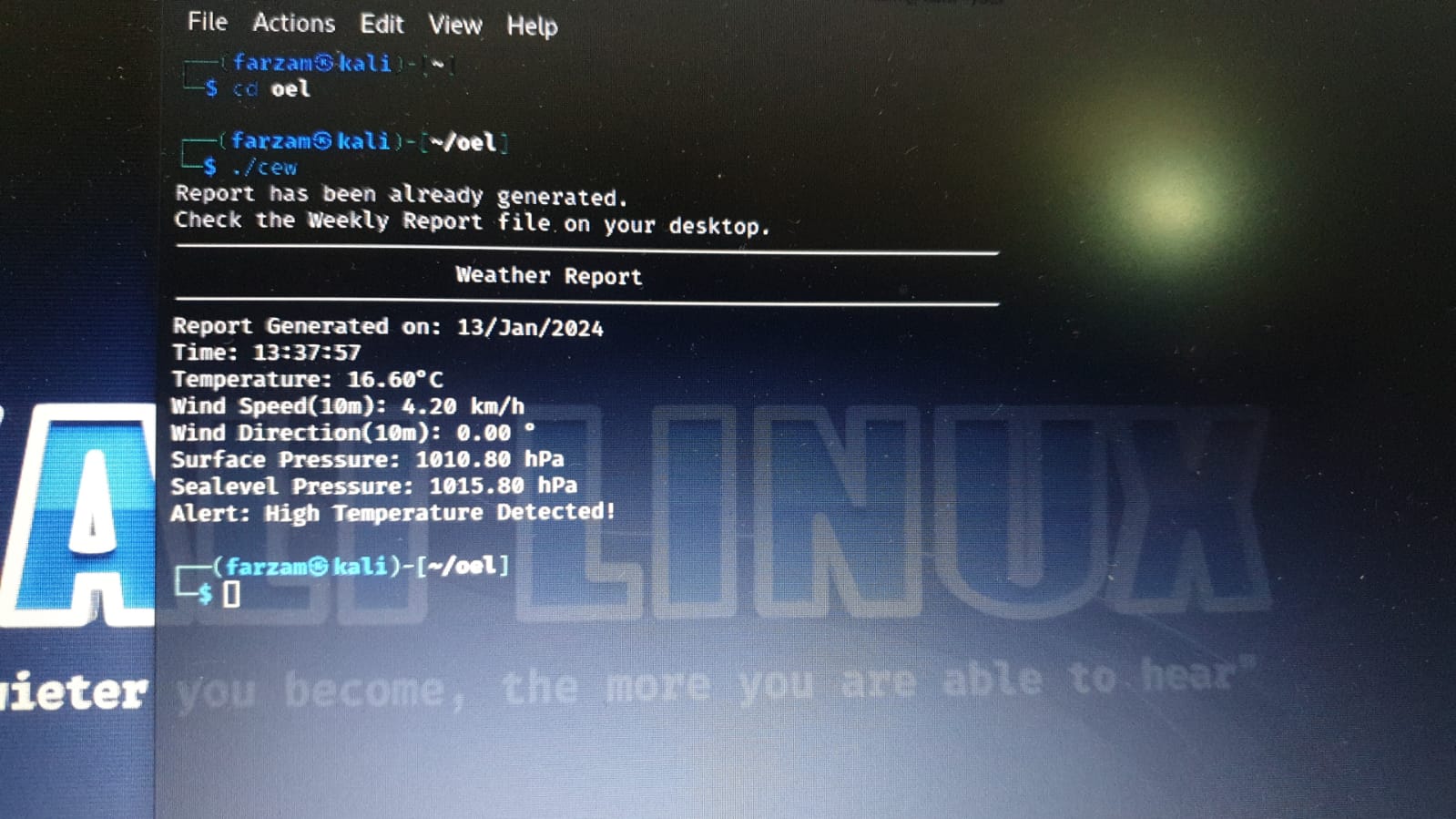
* **2.5: Handling Mechanism:**

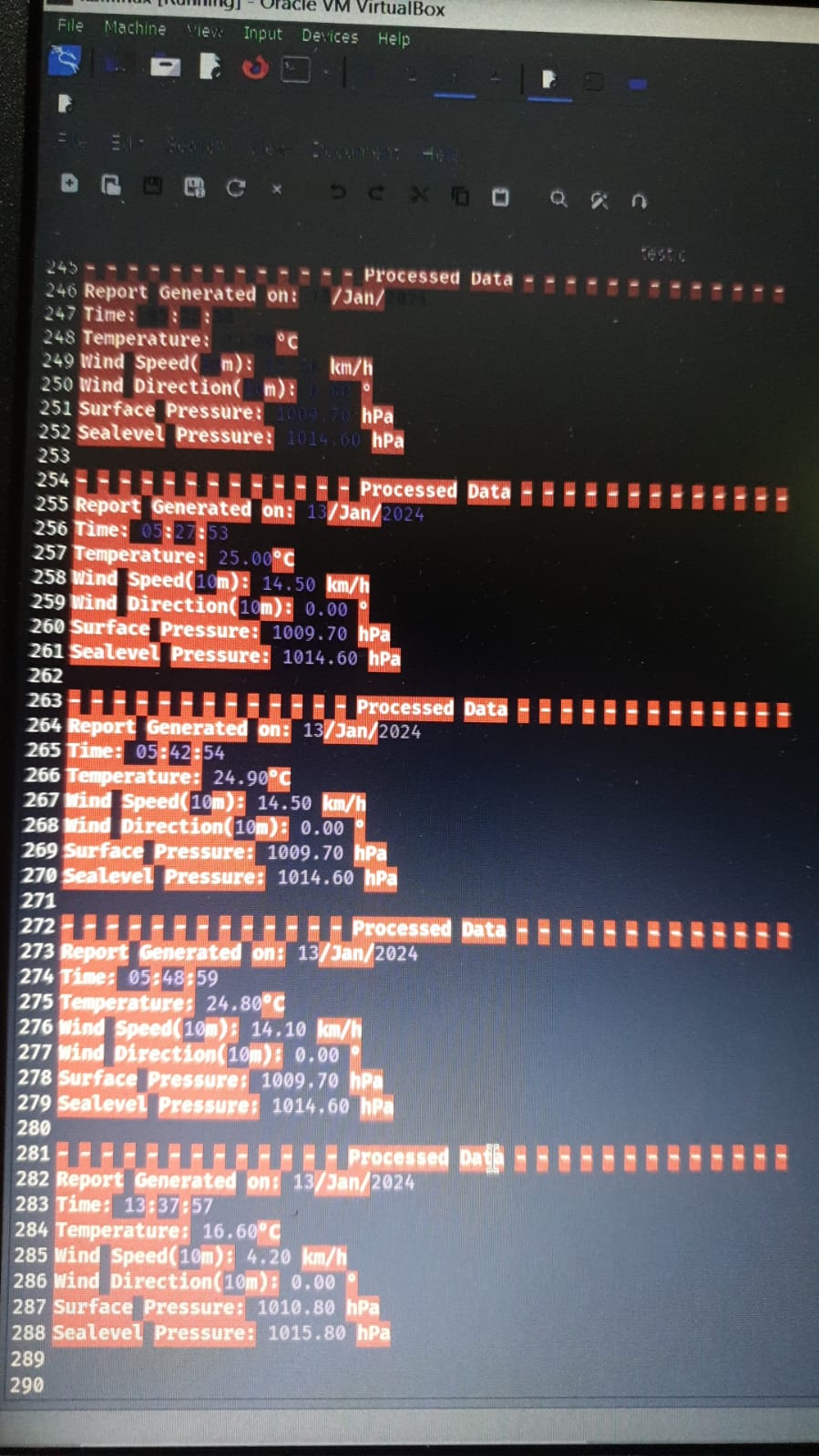
The handling mechanism in the "Handling()" function manages the state of the system based on the current day, ensuring proper functioning and report generation.

* **2.6: Notifications:**

Notifications are displayed on the console for temperature and wind speed alerts. Additionally, email notifications are sent using the curl command, providing enhanced security and authentication.

**OUTPUTS**

****

****

# Conclusion

The weather monitoring system successfully fetches real-time data, processes it, and provides valuable insights through alerts and reports. The implemented features ensure timely actions based on the day of the week, enhancing the overall functionality of the system. Further improvements can include additional anomaly checks and a more sophisticated alerting mechanism.s