A logo of a university of engineering and technology

Description automatically generated

**Software Engineering Lab Manual**

Submitted To:

**Dr. Muhammad Asim Rehmat**

**Ma’am Sana**

Course:

**CMPE-311L : Software Engineering**

Semester:

**5th**

Date of Submission:

**January 16th, 2024**

**Department of Computer Engineering University of Engineering and Technology, Lahore**

**Lab No.1**

**Write the complete problem statement**

**Problem Statement: Air Pollution Monitoring**

Air pollution is a critical environmental concern with far-reaching implications for public health and ecological well-being. The degradation of air quality due to various pollutants poses significant challenges to sustainable living. To address this issue, the proposed project aims to develop an "Air Pollution Monitoring" system, which will provide crucial insights into air quality, helping communities and authorities make informed decisions.

**Background:**

Currently, air pollution is a pervasive problem affecting urban and rural areas alike. Various pollutants, such as particulate matter (PM), nitrogen dioxide (NO2), sulfur dioxide (SO2), carbon monoxide (CO), and ozone (O3), contribute to deteriorating air quality. The existing monitoring systems often lack real-time data, coverage, and accessibility, making it challenging to respond promptly to changing environmental conditions.

**Objectives:**

The primary objective of the "Air Pollution Monitoring" project is to establish a comprehensive and efficient system for monitoring air quality. The key goals include:

**1. Real-time Data Collection:** Implement a network of sensors strategically placed across the target area to collect real-time data on air quality parameters.

**2. Data Analysis and Reporting:** Develop a robust analytical framework to process the collected data and generate meaningful insights. This will involve identifying trends, hotspots, and potential health risks associated with air pollution.

**3. User-Friendly Interface:** Create an intuitive and accessible user interface that allows both the general public and authorities to access air quality information easily. This interface should provide detailed reports, visualizations, and alerts based on the monitored data.

**4. Integration with Existing Systems:** Ensure seamless integration with existing environmental monitoring systems to leverage available infrastructure and enhance the overall effectiveness of air quality management.

**Stakeholders:**

The stakeholders for the "Air Pollution Monitoring" project include government agencies responsible for environmental regulation, local communities, healthcare institutions, and researchers. By involving these stakeholders, the project aims to create a collaborative approach towards tackling air pollution.

**Benefits:**

The successful implementation of the project will lead to:

**1. Improved Public Health:** Timely information about air quality will enable individuals to take precautionary measures, reducing health risks associated with poor air quality.

**2. Informed Decision-Making:** Government agencies can use the collected data to formulate evidence-based policies and regulations to curb air pollution.

**3. Environmental Conservation:** The project will contribute to raising awareness about the impact of human activities on the environment, fostering a sense of responsibility for sustainable living.

**Conclusion:**

The "Air Pollution Monitoring" project seeks to address the pressing issue of air pollution by providing a comprehensive and accessible solution. By focusing on real-time data collection, analysis, and user-friendly reporting, the project aims to empower communities and authorities to take proactive measures in safeguarding the environment and public health. The collaboration of various stakeholders will be pivotal in achieving the project's objectives and ensuring its long-term success.

**Lab No.2**

**Write the software requirement specification document**

**Software Requirement Specification for Air Pollution Monitoring**

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to provide a detailed specification for the development of the "Air Pollution Monitoring" software system. It outlines the requirements, functionalities, constraints, and interfaces essential for the successful implementation of the project.

**1.2 Document Conventions**

- Main headings are **black.**

- Sub-headings are **blue**.

- Sub-Sub-headings are ***italic black***.

**1.3 Intended Audience**

This document is intended for the development team, stakeholders, and any parties involved in the design, implementation, and testing of the "Air Pollution Monitoring" system.

**1.4 Contact Information/SRS Team Members**

- Project Manager: [Name]

- Technical Lead: [Name]

- Developer: [Name]

- Tester: [Name]

**1.6 References**

<https://www.googleadservices.com/pagead/aclk>

<https://www.googleadservices.com/pagead/aclk>

**2. Overall Description**

**2.1 Product Perspective**

The "Air Pollution Monitoring" system is a standalone application that interfaces with a network of environmental sensors. It is not dependent on other systems but may integrate with existing environmental monitoring infrastructure.

**2.2 Product Functions**

- Real-time data collection from environmental sensors.

- Data analysis and interpretation.

- User-friendly interface for accessing air quality information.

- Integration with existing environmental monitoring systems.

- Alerting mechanism for potential health risks.

**2.3 User Classes and Characteristics**

- Public Users: Access air quality information through the user interface.

- Administrators: Manage system settings, monitor data, and receive alerts.

**2.4 Operating Environment**

The system will be designed to operate in urban and rural environments, utilizing a network of sensors strategically placed in the target area.

**2.5 User Environment**

Users will access the system through a web-based interface, compatible with commonly used web browsers.

**2.6 Design/Implementation Constraints**

- The system must comply with environmental regulations.

- Sensor deployment and maintenance costs should be considered.

- The system should be scalable to accommodate additional sensors and users.

**2.7 Assumptions and Dependencies**

- The availability of a reliable network infrastructure for sensor communication.

- Sensor accuracy and reliability are assumed within acceptable limits.

**3. External Interface Requirements**

**3.1 User Interfaces**

The system will feature a web-based user interface providing real-time air quality data, visualizations, and alerts.

**3.2 Hardware Interfaces**

- Sensor devices for data collection.

- Standard computer systems for server hosting.

**3.3 Software Interfaces**

- Database management system for data storage.

- Web server software for hosting the user interface.

**3.4 Communication Protocols and Interfaces**

Communication between sensors and the system will utilize standard protocols, such as HTTP or MQTT.

**4. System Feature**

**4.1 System Feature A - Real-time Data Collection and Analysis**

***4.1.1 Description and Priority***

The system must collect real-time data from environmental sensors and analyze it to provide accurate air quality information. Priority: High.

***4.1.2 Action/Result***

Upon sensor data reception, the system will process and analyze it, updating the user interface with the latest air quality metrics.

***4.1.3 Functional Requirements***

- Receive data from environmental sensors.

- Implement algorithms for data analysis.

- Update the user interface with real-time air quality information.

**4.2 System Feature B - User Interface**

***Description and Priority:***Provide an intuitive and accessible web-based interface for users to access air quality information. Priority: Medium.

***Action/Result:*** Users can log in, view real-time air quality data, historical trends, and receive alerts.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

- The system should handle concurrent user access.

- Data processing time should not exceed 5 seconds.

**5.2 Safety Requirements**

- The system should not compromise the safety of users or the environment.

- Alerting mechanisms for potential health risks should be reliable and timely.

**5.3 Security Requirements**

- User authentication for access to sensitive data.

- Data encryption during transmission.

**5.4 Software Quality Attributes**

- ***Reliability:*** The system should operate continuously with minimal downtime.

- ***Scalability:*** The system should scale to accommodate additional sensors and users.

- ***Maintainability:*** Codebase should be well-documented and easily maintainable.

**5.5 Project Documentation**

- Regularly updated design documentation.

- User manuals and guides.

**5.6 User Documentation**

- Comprehensive user guide.

- Troubleshooting documentation.

**Appendix A: Terminology/Glossary/Definitions List**

- AQI: Air Quality Index

- PM: Particulate Matter

- NO2: Nitrogen Dioxide

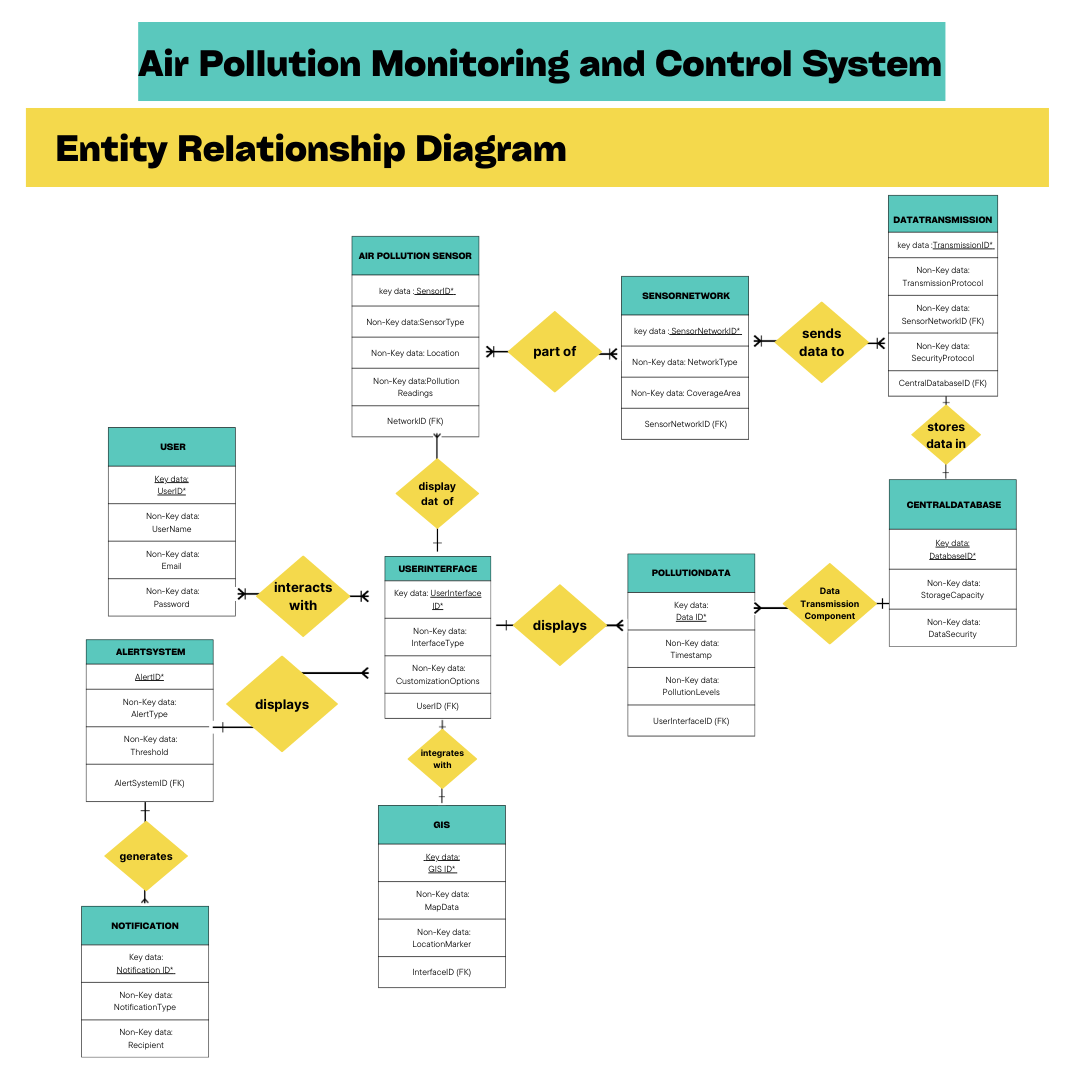
- SO2: Sulfur Dioxide

- CO: Carbon Monoxide

- O3: Ozone

**Lab No.3**

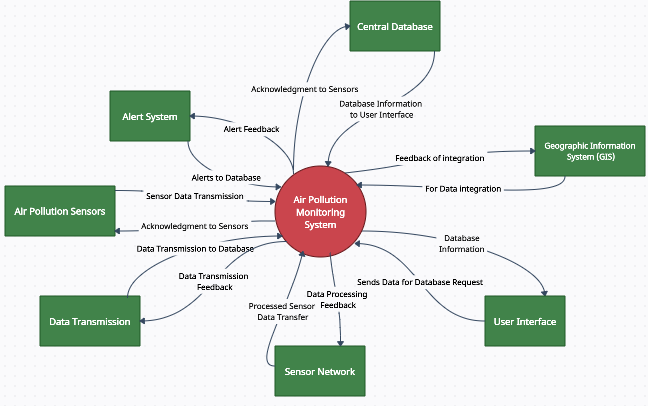
**Draw the entity relationship diagram**

****

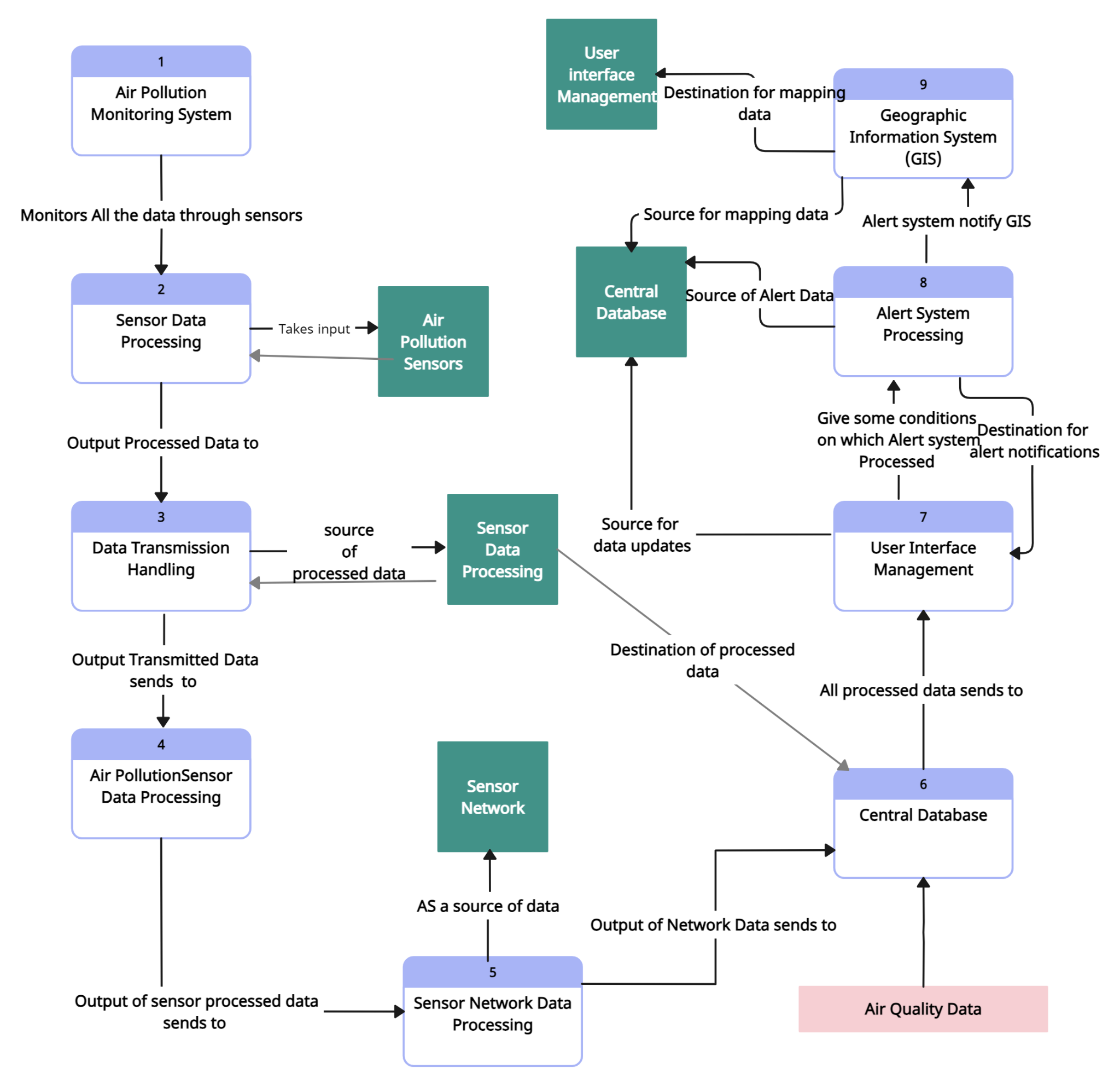
**Lab No.4**

**Draw the data flow diagrams at level 0 and level 1**

**DFD LEVEL 0**

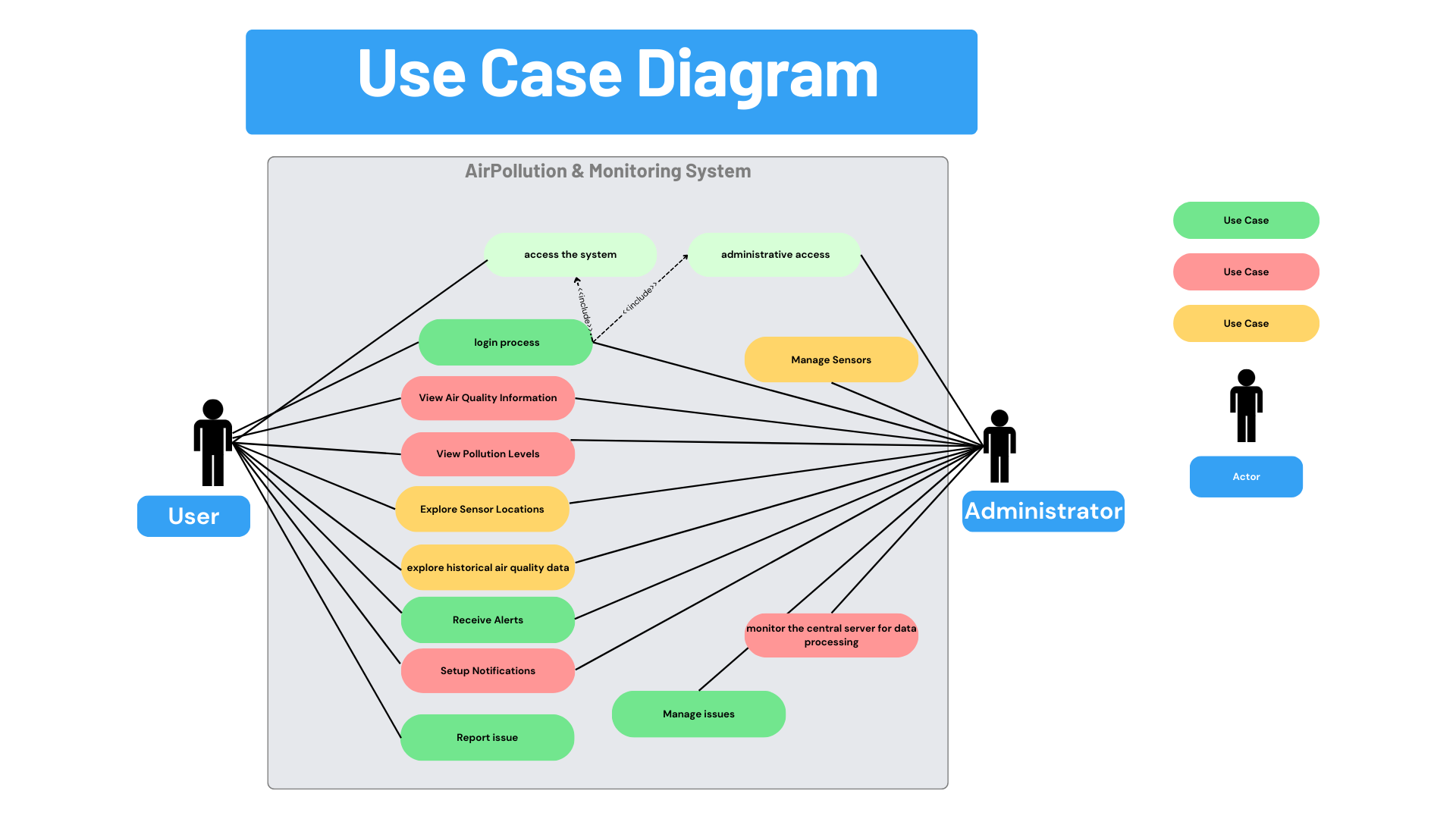
****

**DFD LEVEL 1**

****

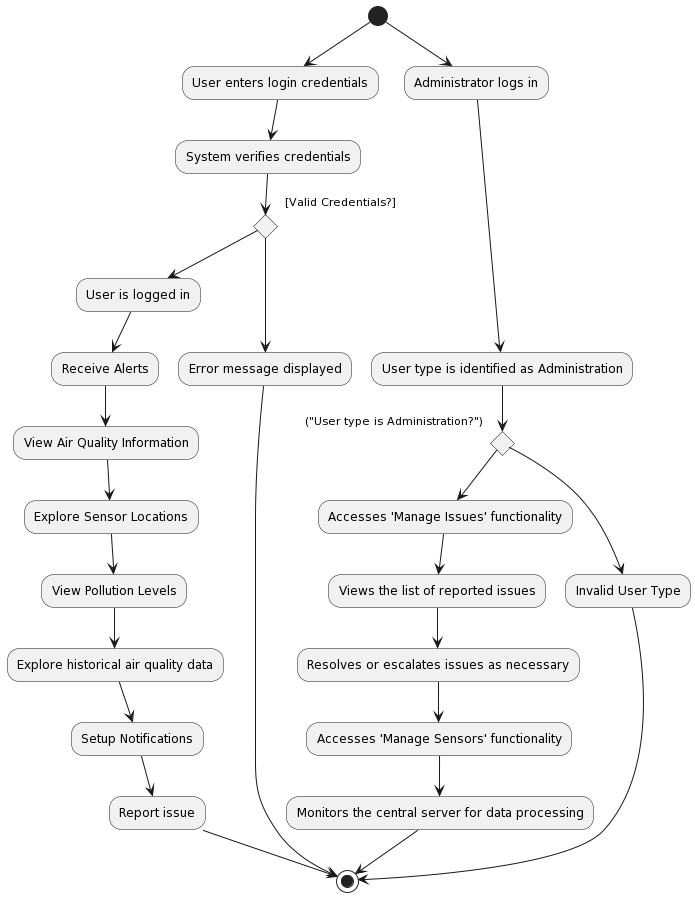
**Lab No.5**

**Draw Use Case Diagram**

****

**Lab No.6**

**Draw Activity Diagram of All Use Cases**



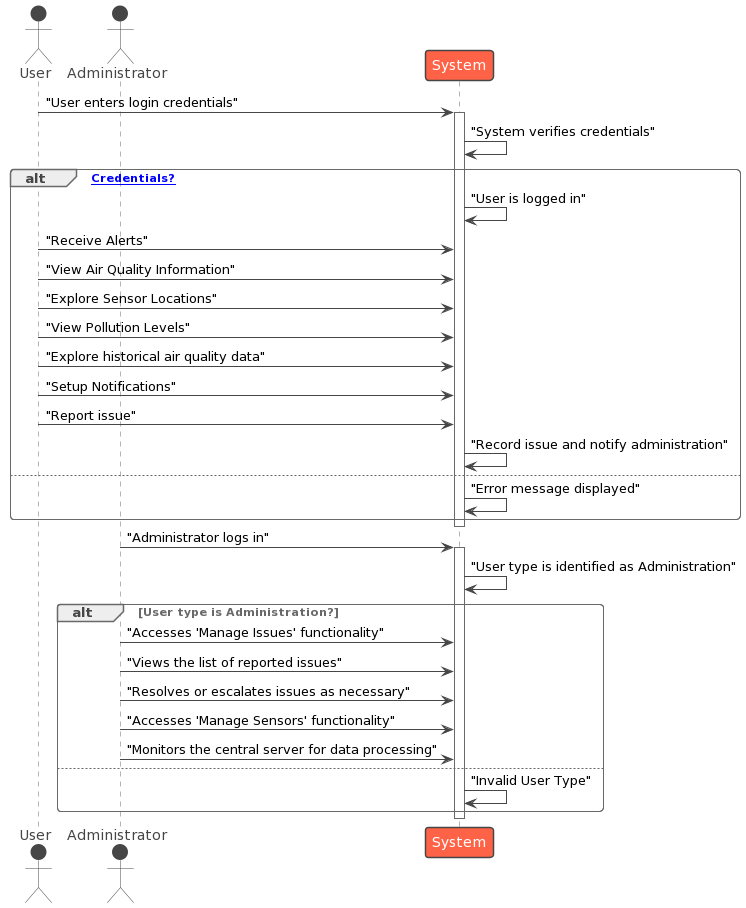
**Lab No.7**

**Draw State Chart Diagram of All Use Cases**

****

**Lab No.8**

**Draw Sequence Diagram of All Use Cases**



**Lab No.9**

**Collaboration Diagram**

monitors air quality data

requests historical data

explores historical air quality data

receives alerts

selects time period

requests sensor location data

explores sensor locations

requests air quality information

manages sensors (add, remove, modify)

manages issues (view, resolve)

monitors central server

sends alerts to user's notifications method

configures notification preferences

**AIR QUALITY INFORMATION**

**CENTRAL SERVER**

**SENSORS**

stores notification preferences

initiates login process

**USER**

**LOGIN PROCESS**

**ADMINISTRATOR**

**Lab No.10**

**Assign objects in sequence diagram to classes and make class diagram.**

