VIVEKANAND EDUCATION SOCIETY'S
INSTITUTE OF TECHNOLOGY
Department of Information Technology
Academic Year: 2024-25

FINAL YEAR BE - MINI PROJECT (IoE) PROPOSAL (2024 - 25)

Team Members

Group No.	Names	Roll No.
10	Yajat Bhasin	5
	Kaushik Kotian	29
	Mohit Mandhyani	34
	Swapnil Mulani	67

Name of Guide: - Abhay Kshirsagar

Project Title: - Weather Monitoring System

VIVEKANAND EDUCATION SOCIETY'S
INSTITUTE OF TECHNOLOGY
Department of Information Technology
Academic Year: 2024-25

Introduction:

The Weather Monitoring System project, built using NodeMCU ESP8266, provides a comprehensive tutorial on developing a real-time weather data collection and monitoring solution. This system, ideal for IoT applications, is designed to track environmental conditions and display data for analysis. With step-by-step instructions, the project is accessible to beginners and tech enthusiasts, offering practical insights into home automation and environmental tracking. By utilizing NodeMCU ESP8266 for Wi-Fi connectivity, it demonstrates how to create an efficient weather monitoring tool that promotes sustainability and proactive decision-making.

Literature Review:

A real-time weather monitoring system for a smart home and data is sent to cloud and data using telemetry transport protocol. There are many local weather stations around the globe, which can be collected from the authorities and this data can be visualized for the farmers to view real-time weather data. Cloud services can be used to store sensor data and later the information can be shown on web-app. Modules like NodeMCU client can be used. Microcontroller like Arduino and low-cost computers like Raspberry Pi can be used to build weather stations. This helps developers to build their own gadgets.

Hardware/Software Requirements:

- **Microcontroller/Development Board:** ESP8266.
- **Sensors:**
  - Rain Sensor
  - DHT 11 Sensor
  - Light Sensor (e.g., LDR or Photoresistor)
- **Actuators:**
  - LCD Display
  - Buzzer
- **Power Supply:** Suitable for powering the heating element and other devices.
- **Cloud Platform:** Blynk App
- **Programming Languages:** Python, C/C++ for microcontroller coding, and JavaScript for the web interface.
- **User Interface:** Web or mobile app developed using React, Angular.

Methodology:

The methodology for developing the Weather Monitoring System using NodeMCU ESP8266 involves several key stages:

1. Sensor Integration: Connecting weather sensors such as temperature, humidity, and pressure sensors to the NodeMCU for accurate environmental monitoring.

2. Data Collection and Processing: Implementing logic for gathering real-time weather data from the sensors and processing it on the microcontroller.

3. Cloud Connectivity: Establishing a connection to a cloud platform or server to enable remote data monitoring and storage for future analysis.

4. User Interface Development: Creating a web or mobile app interface to display real-time weather data and allow users to access historical records.

5. Testing and Calibration: Testing the system in different environmental conditions and calibrating the sensors for accurate and reliable data collection.

6. Optimization: Optimizing data transmission and sensor performance to enhance the overall efficiency and responsiveness of the system.

Implementation:

Here are the steps to implement your weather monitoring system:

1. **Set Up the NodeMCU in Arduino IDE:**

Correctly describes the setup in Arduino IDE, including installing necessary libraries and configuring NodeMCU.

2. **Connect the Sensors and LCD:**

Properly includes wiring up sensors (DHT11, rain sensor, LDR) and connecting the LCD display with I2C module, which is an essential part of the sensor integration process.

3. Write the Code:

The code section outlines key functionalities like reading sensor data, initializing the Blynk app, and displaying data on the LCD and Blynk app, covering the data collection and processing phase.

4. Upload Code to NodeMCU:

Uploading the code and compiling it through Arduino IDE fits with the typical project workflow.

5. Set Up Blynk App:

Using the Blynk app for remote monitoring ensures cloud connectivity and provides a user interface, as expected from the methodology.

6. Test the System:

Testing and calibrating the system is essential for ensuring data accuracy and reliable performance.

7. Finalize:

Adding features like alerts or triggers aligns with optimization and expanding functionality.

VIVEKANAND EDUCATION SOCIETY'S
INSTITUTE OF TECHNOLOGY
Department of Information Technology
Academic Year: 2024-25

Plan of Implementation

MONTH	PLAN
July 2024 August 2024	Project selection
	Project initiation and research
September 2024	Data Collection and Preparation (Database Management)
	User Interface Design
	Task logic implementation
	Error Handling and user personalization
October 2024	Integration and deployment
	Testing
	Final Submission (Project Report, PowerPoint Presentation)

Applications: The Weather Monitoring System using NodeMCU ESP8266 offers a wide range of applications for both personal and professional use, enabling real-time data monitoring for enhanced decision-making and environmental awareness. Here are some key applications:

1. **Real-Time Weather Monitoring :**

The system provides accurate, real-time monitoring of weather parameters such as temperature, humidity, and pressure, enabling users to stay informed about current weather conditions.

2. **Environmental Data Analysis :**

Collected data can be analyzed to track weather patterns, climate changes, and environmental conditions over time. This is especially useful for weather enthusiasts, researchers, and environmental monitoring agencies.

3. **Home Automation Integration :**

The system can be integrated with home automation devices to adjust HVAC systems based on environmental conditions, optimizing indoor comfort while conserving energy.

4. **Agriculture and Farming :**

Farmers can use the system to monitor microclimate conditions in their fields, helping them make data-driven decisions about irrigation, planting, and harvesting, leading to more efficient crop management.

5. **Smart City Applications :**

In smart cities, the system can be deployed to monitor local weather conditions, providing valuable data for traffic management, public safety, and infrastructure maintenance.

References :

1. WEATHER_MONITORING_SYSTEM_USING_INTERNETOF_THINGS_IOT
2. IOT BASED WEATHER MONITORING SYSTEM

VIVEKANAND EDUCATION SOCIETY'S
INSTITUTE OF TECHNOLOGY
Department of Information Technology
Academic Year: 2024-25

Signature of the Guide: -