National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

CS-335: INTERNET OF THINGS

BESE-9B

Project Report

IOT based Live Weather Station System to monitor Temperature, Humidity and Pressure

Submission Date: 3rd January 2022

Instructor: Rafia Mumtaz

Group members:

Name	CMS
Syed Babar Tameez	252769
Fatima Khalid	256611
Ghazala Bibi	261258
Safi ur Rehman	241984
Anber Arif	264973

1. Brief introduction of the project.

Our project is IoT based Live Weather Station Monitoring Using NodemCU ESP8266**.** We will interface DHT11 Humidity & Temperature Sensor and BMP280 barometric Pressure Sensor with NodeMCU ESP8266-12E Wifi Module**.**

A **weather station** is a device that collects data related to the weather & environment using different sensors.

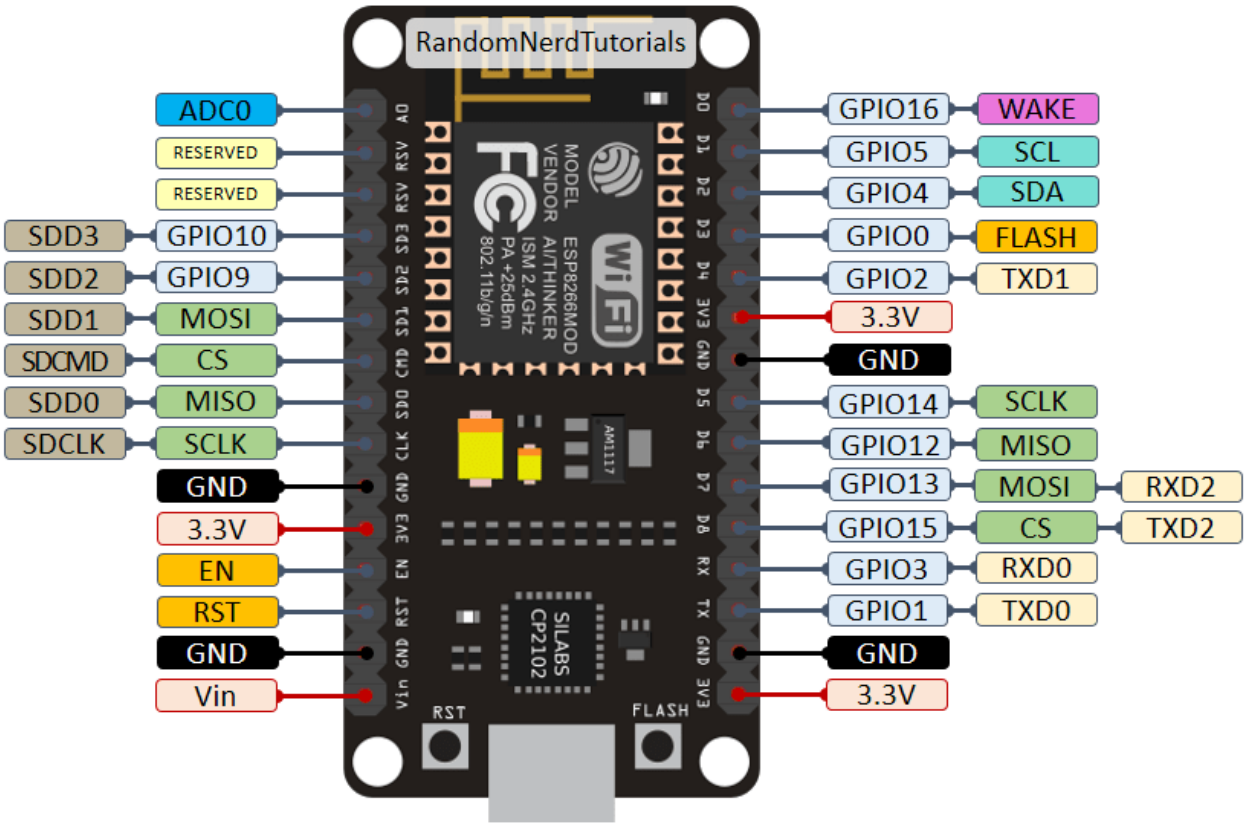
Objectives

- To measure humidity
- To measure temperature
- To measure barometric pressure
- To visualize the data using a mobile app (Blynk).

2. Details of Sensors and related electronic equipment.

- **NodeMCU ESP 8266 Wi-Fi Module**

It is an open-source LUA based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems and hardware which is based on the ESP-12E module. NodeMCU has 128 KB RAM and 4MB Flash memory to store data and programs. We can establish a Wi-Fi connection with just few lines of code and can define input/output pins according to the needs just like Arduino, turning ESP8266 into a web server. It is a Wi-Fi equivalent of ethernet module.



Node-MCU-ESP 8266 pinout diagram

- **Features**

It provides following features.

- Programable Wi-Fi module.
- Arduino-like hardware IO.
- Event-driven API for network applications.
- Wi-Fi networking – can be used as access point, host a web server, connect to internet to fetch, or upload data.
- Can be programmed with simple and powerful Lua programming language or Arduino IDE.
- Small sized module to fit smartly inside our IoT project.

- **NodeMCU ESP 8266 Pinout Configuration**

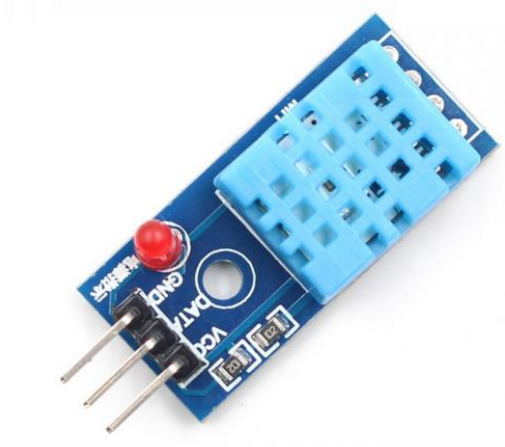
Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	**Micro-USB:** NodeMCU can be powered through the USB port **3.3V:** Regulated 3.3V can be supplied to this pin to power the board **GND:** Ground pins **Vin:** External Power Supply
Control Pins	EN, RST	The pin and the button reset the microcontroller
Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	Four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	Two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

➢ **Technical Specifications**

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins: 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4MB
- SRAM: 64KB
- Clock Speed: 80MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna

- **DHT-11(Temperature and Humidity Sensor)**

  DHT-11 is a Temperature and Humidity Sensor, which generates calibrated digital output. It can be interface with any microcontroller and get instantaneous results. It is a low-cost humidity and temperature sensor which provides high reliability and long-term stability.



- **DHT11 Pinout Identification and Configuration**

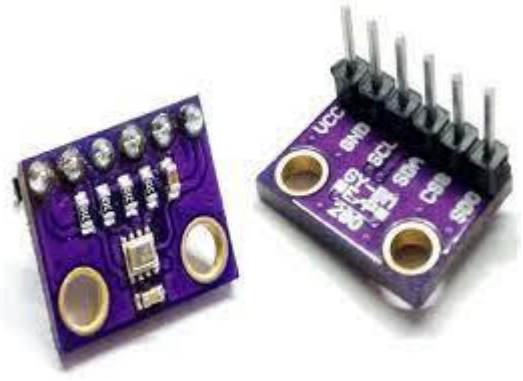
Sr. No.	Pin Name	Description
1	VCC	Power Supply 3.5 V to 5.5V
2	Data	Outputs both Temperature and Humidity through Serial Data
3	Ground	Connected to the ground of the circuit

- **DHT11 Technical Specifications:**

  - Operating Voltage: 3V to 5.5V
  - Operating Current: 0.3mA (measuring), 60uA (standby)
  - Output: Serial Data
  - Temperature Range: 0-50 °C
  - Humidity Range: 20% to 90%RH
  - Resolution: 1
  - Accuracy: ±2C and ±5%RH

- **BMP-280 (Barometric Pressure Sensor)**

  It is an absolute barometric pressure sensor. Its small dimensions and its low power consumption allow for the implementation in battery-powered devices such as mobile phones, GPS modules or watches. It is based on Bosch's proven piezo-resistive pressure sensor technology featuring high accuracy and linearity as well as long-term stability and high EMC robustness. It is optimized in terms of power consumption, resolution, and filter performance.



- **Pins Configuration:**

Pin	Name	Function
1	GND	Ground
2	VCC	Power pin to connect 3.3V power supply
3	SCL	Serial Clock pin for I2C interface
4	SDA	Serial Data pin for I2C interface
5	SDO	Serial Data Output pin
6	CSB	Select pin for I2C or SPI interface

- **BMP280 Technical Specifications:**

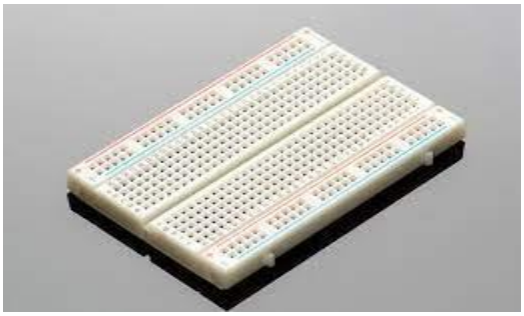
  - Operation range: Pressure -> 300-1100 hPa
  - Absolute accuracy (950-1050 hPa, 0…+40°C): ~ ±1 hPa
  - Relative accuracy: ± 0.12 hPa (typical) equivalent to ±1 m
  - Average typical current consumption (1 Hz data rate): 3.4 µA @ 1 Hz
  - Average current consumption (1 Hz data refresh rate): 2.74 µA, typical(ultra-low power mode)
  - Average current consumption in sleep mode: 0.1 µA
  - Average measurement time: 5.5 msec(ultra-low power preset)
  - Resolution of data: Pressure-> 0.01 hPa (< 10 cm)
  - Interface: I²C and SPI
  - Package dimensions: 11.5mm x 15mm

- **Jumper Wires**

  Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards to make it easy to change a circuit as needed.

- **Breadboard**

  It is used to build and test circuits quickly before finalizing any circuit design. The breadboard has many holes into which circuit components like ICs and resistors can be inserted. It has strips of metal which run underneath the board and connect the holes on the top of the board. Top and bottom row of holes are connected horizontally while the remaining holes are connected vertically.



3. Detailed description of a specific communication technology and possible network topologies if the system needs to be scaled up.

WiFi has been used as a communication technology at physical level. WiFi is required to operate the Blynk app. As our system requires high bandwidth and low latency, WiFi is most suitable for it. WiFi is uniquely placed to support broadband and narrowband IoT applications from a common platform that can work at varying levels of power consumption and signal range.

- **Network topologies**

  - **Present topology**

    As we are using WiFi in our system and it has a star topology, so it means our system is based on star topology.

  - **Future topology**

    If we scale up our system then we can use ZigBee as our communication technology protocol at physical level as it consumes low power and able to connect many devices at a moment, moreover it has low network join time of 3ms. ZigBee consists of all three topologies, i.e., star, mesh and tree topology but mesh is more suitable as Mesh topology is the most effective since data can be sent to the target in more ways, it can transfer data to a destination even if parts of the system are lost.

4. Perform data analysis, create dashboard for data visualization for displaying infographics and select specific system interface.

Blynk app is used to monitor temperature, pressure and humidity. Blynk app is an easy to use Android and IOS based application, helps in controlling Arduino, ESP8266, Raspberry Pi and others for the purpose of IoT. Blynk app is a digital dashboard, where you can build a graphic interface for your project by simply dragging and dropping widgets. The best part of Blynk app is that, it is not bounded with some specific board or platform. It is a hardware-agnostic IoT platform with white-label mobile apps, private clouds, device management, data analytics, and machine learning.

- **Create a Blynk Account**

After downloading the Blynk App, we created a New Blynk account with a valid email address for the following reasons.
An account is required to save user's projects and have access to them from multiple devices from anywhere in the world.
It's also a security measure.
The users can always set up their own Private Blynk Server and have full control.

- **Create a New Project**

After successfully logged into the account, the next step is to start by creating a new project.

- **Choose Hardware**

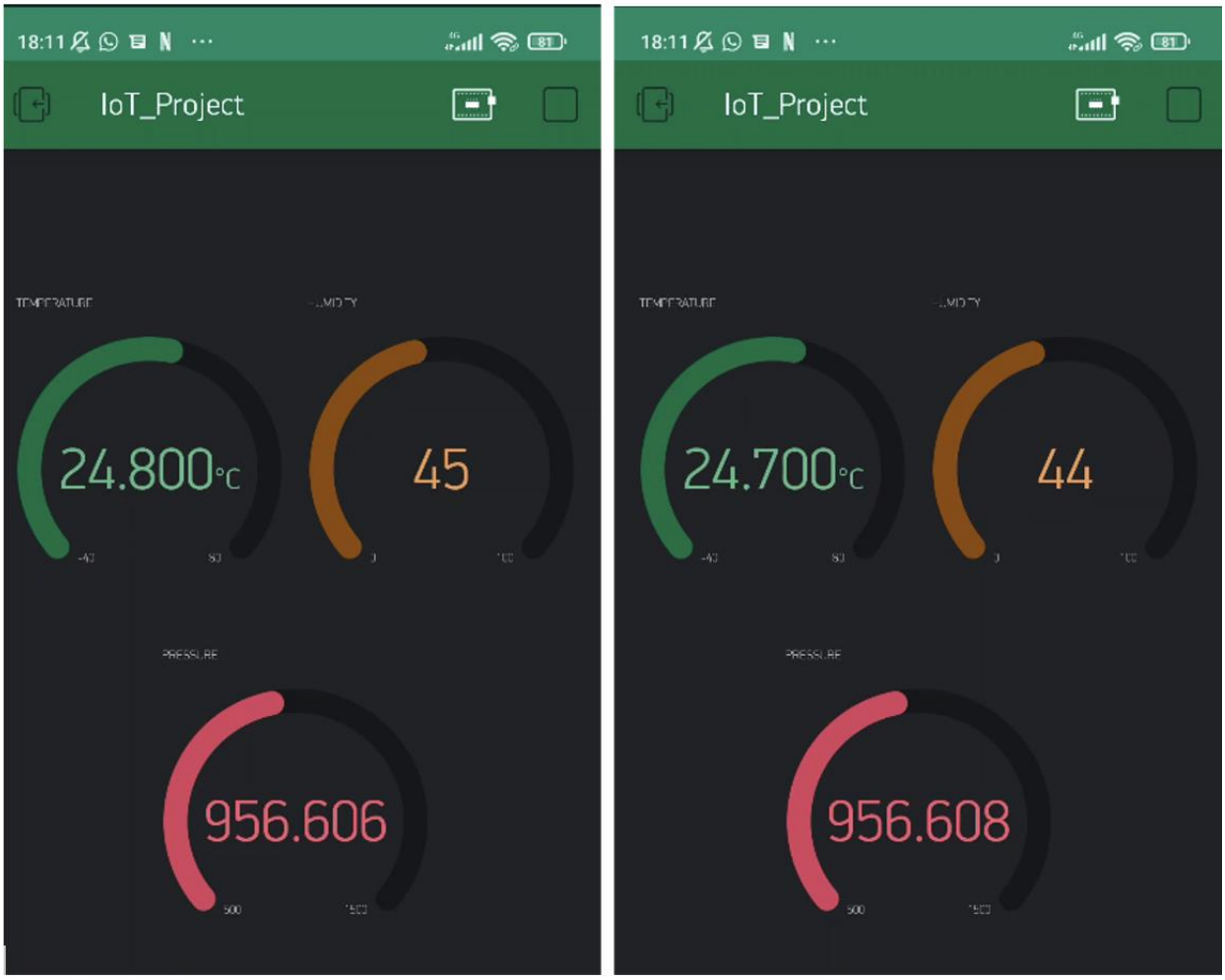
User needs to select the hardware model to be used. Blynk supports more than 400 boards. For our system we used Node MCU ESP 8266.

- **Auth Token**

Auth Token is a unique identifier for the user which is needed to connect their hardware to their smartphone.

- **Add a Widget**

As the project canvas was empty, it required to add widgets or blocks for the project. For that purpose we tapped anywhere on the canvas to open up the widget box. All the available widgets are located here. We picked and placed 3 gauges to display temperature, pressure and humidity.



- **Run the project**

After done with the Settings, we pressed the PLAY button. This switched the project from EDIT mode to PLAY mode where we interacted with the hardware.

5. Detailed description of power consumption of every sensor should be included and the approach that can be used for the optimal consumption of power should be described.

- **DHT-11 Temperature and Humidity Sensor**
  - **Power Consumption**
    It consumes power of 3-5V and draws max current of 2.5mA.
  - **Approach for optimal Consumption of Power**
    The sensor operates in sleep/standby mode consuming 60uA, compared to 0.3mA when making measurements.
- **BMP-280 (Barometric Pressure Sensor)**
  - **Power Consumption**
    It has low power consumption of 2.7 µA @1Hz.
  - **Approach for optimal Consumption of Power**
    The sensor operates in sleep/standby mode consuming 0.1uA, compared to 2.7uA – 3.4uA when making measurements.

6. Describe Application Requirements.

- Blynk app is used to display the infographics
- Arduino-IDE was used for burning the code on chip, hence C/C++ used in writing that program.

7. Describe tools and technologies.

- Node MCU ESP 8266 WiFi Module
- DHT-11 (Temperature and Humidity Sensor)
- BMP-280 (Barometric Pressure Sensor)
- Jump Wires
- Breadboard
- Blynk App
- Arduino-IDE (Programming language: C/C++)