

Monitoring Patient Based-Home

Hussain Laith Abbas, Taher Falah Abd, Taha qusay Talib, Mustafa Hadi kadhim
Bio Medical Engineering Department, University of Technology

Dr. Zeyad Qasim Habeeb
Supervisor

Chapter One: Introduction

1. Introduction

Remote patient monitoring (RPM) is a form of telehealth, or healthcare delivered remotely. The popularity of telemedicine has been on the rise since the outbreak of the pandemic, it brings in new models of care along remote patient monitoring (RPM) of key health metrics to improve chronic care management next decade will see an impressive increase in remote patient monitoring (RPM) devices, at a growth rate of 12.5 percent annually. The trend is quite predictable considering the cumulative effects of the aging population, the high cost of in-patient care, and enormous pressure on hospitals caused by COVID-19.

Healthcare providers are already using more equipment for remote monitoring than ever before. To free up beds, many hospitals started tracking vital signs and symptoms of both coronavirus and non-coronavirus patients from a distance.

RPM practices were supported by authorities. The US Centers for Medicare and Medicaid Services (CMS) expanded the list of telehealth services eligible for reimbursement. And the US Food and Drug Administration (FDA) authorized the use of non-invasive devices designed for hospitals in home settings.

2. Objectives

1. Reduce cost by shortening the patient's stay in the hospital.
2. Improving the psychology of patients through quality in his family and in his home.
3. Make easy for people to check their main healthy functions.
4. Easy use for any user.
5. Minimize the size of the device.

3. Project's Benefits

1. Organizational Mobility.
2. Less Travel for Home Health Caregivers.
3. Streamlined Communications with the Patient's Medical Network.
4. Home Healthcare Device and Technology Support.

4. Project outlines

1. Study of microcontrollers, sensors, and how to use them in the project.
2. Create an integrated circuit containing DHT11 (humidity and temperature sensor), kAD8232 ECG Sensor Module, and MLX90614 (Infrared thermometer).

3. Designing a device that analyzes the electrical signal of the heart and whether this signal is normal or not.
4. Availability of the device at home and sending image of ECG to the doctor to make the diagnosis.
5. Team work and how to divide work among team members.

Chapter Two: Literature Review

1. Literature Survey

With advances in display technologies through the 1990s and early 2000s, and the advent of the touch screen, patient monitor systems have become both easier to use and to transport! how technology has improved the efficiency of medical professionals across the globe [1].

In the olden days, the patients' health was monitored by catching his /her hand by checking their pulses. As time passed on and the technology for monitoring health got introduced, the quality of measuring and understanding the health conditions got better. The development in the technology grew to such an extent that nowadays by wearing a small device, a patient's health is getting monitored [2]. Few projects are concerned with developing wearable wrist-worn belts, while others have developed based for monitoring individual patients during daily activity, at home, or in hospital. A liker, J. A. Ward et.al has developed a wearable medical monitoring and alert system aimed at people at risk from the heart and respiratory diseases [3]. The system combines multi-parameter measurement of vital signs (Heart rate and body temperature) and also show air temperature and humidity to see if the patient is in an optimal environment

Doctors and nurses alike are now able to monitor and report on patient vitals effortlessly, and with portable patient monitoring systems like the Philips IntelliVue X3 – moving patients from one part of the facility to another has become much less of a hassle.

One can only imagine where patient monitoring systems are destined to venture next – what with the great advances in holographic technology and microchip capabilities of recent years [1].

The following are some related works:

2. IoT ECG Monitoring with AD8232 ECG Sensor & ESP8266

Heart diseases are becoming a big issue for the last few decades and many people die because of certain health problems. Therefore, heart disease cannot be taken lightly. So there should be a technology that can monitor the heart rate and heart behavior of the patient regularly. By analyzing or monitoring the ECG signal at the initial stage the various heart disease can be prevented.

This is the reason why I am presenting you with this great IoT project. In this project, I will show you how you can interface AD8232 ECG Sensor with NodeMCU ESP8266 Board and monitor the ECG Waveform on Serial Plotter Screen. Similarly, you can send the ECG waveform over the IoT Cloud platform and monitor the signal online from any part of the world using the PC or simply using the Smartphone. There is no need for staying in the Hospital to monitor heart activity/behavior just because you can monitor it online from anywhere. Thus it can be said advancement in Patient Health Monitoring System.

The IoT platform that I am gonna use here is Ubidots. Ubidots is an IoT Platform empowering innovators and industries to prototype and scale IoT projects to production. Use the Ubidots platform to send data to the cloud from any Internet-enabled device. You can then configure actions and alerts based on your real-time data and unlock the value of your data through visual tools [4].

3. IOT Patient Health Monitoring Project

Monitoring your beloved ones becomes a difficult task in modern day life. Keeping track of the health status of the your patient at home is a difficult task. Specially old age patients should be periodically monitored and their loved ones need to be informed about their health status from time to time while at work. So we propose an innovative system that automated this task with ease. Our

system puts forward a smart patient health tracking system that uses Sensors to track patient health and uses internet to inform their loved ones in case of any issues. Our system uses temperature as well as heartbeat sensing to keep track of patient health. The sensors are connected to a microcontroller to track the status which is in turn interfaced to an lcd display as well as wifi connection in order to transmit alerts. If system detects any abrupt changes in patient heartbeat or body temperature, the system automatically alerts the user about the patients status over IOT and also shows details of heartbeat and temperature of patient live over the internet. Thus IOT based patient health tracking system effectively uses internet to monitor patient health stats and save lives on time [5].

4. HOME AUTOMATION ON ESP8266

Home automation is commonly called smart home. It involves the control and handle the things like light, fan, door, alarming for certain remainders, security purposes etc. All the things are connected to the internet and all of them can accessed at any place and any time. Most popular protocol communication for the products include X10, Ethernet, ZigBee and Z-Wave, or other protocols are all incompatible with other. The web server is simultaneously updated by sensing the status of the things which are connected to the network. The status of the appliances is controlled by the switch it either ON or OFF using the computer technology. It also provides the security, energy efficient, and ease of use, hence it is adopted more. It also helps by providing to control and monitoring on web browser. The main objective of the project is to help handicapped people and aged people by alerting in the critical situations. All the devices can be used in our own sitting place itself. The problem overcome by this paper is about that home automation is generally implemented by using Bluetooth through our PC. Pin check algorithm is used to implement this set-up by using the cable network other than the wireless communication. The device ESP8266, which is the embedded device used to access the cloud. The home automation using Bluetooth with the help of PC has implemented by N.Srikanthan .The range is limited by using the Bluetooth device. The devices are controlled by the telephone and with PIC controlling remote device by Hasan. The IOT devices uses different types of protocol. MQTT (Message Queuing Telemetry Transport) is one of the useful and advance. Because it is easy to implement, open, lightweight, low bandwidth protocol. We can also use PIR sensor and proximity. Where the devices in IOT is used for controlling or nominating the devices where all of them are far away from this. MQTT and TCP protocols are also used to implement the ESP8266 Wi-Fi module. The potential IOT applications develop the environment that covers all together the applications, command, control and routing process and security of the node and system. All the IOT devices include various objects like personal computer, smart phones, tablets, which gives the communication between the things and people and also the things between them. To reduce the need for the home intervention, we are used to control the home automation through the use of control systems [6].

5. Smart Health Care Monitoring System Based on IoT

This project is significant in various ways because in today's world, every day many lives are affected because the patients are not timely and properly operated on. Also, real-time parameter values are not efficiently measured in clinics as well as in hospitals. Sometimes it becomes difficult for hospitals to frequently check patients' conditions. Also, continuous monitoring of ICU patients is very difficult. To deal with these types of situations, our system is beneficial. Our system is designed to be used in hospitals and homes also for measuring and monitoring various parameters like temperature, ECG, heart rate, blood pressure. The results can be recorded using Arduino. Also, the doctors can see those results on the android app. The system will also generate an alert notification which will be sent to the doctor. Our system is useful for monitoring the health system of every person by easily attaching the device and recording it. In which we can analyze patients' conditions through their past data, we will recommend medicines if any emergency occurred through symbolic A.I [7].

➤ Software apps and online services

1. Android Studio
2. Arduino IDE
3. Google Firebase
4. Hand tools and fabrication machines
5. 10 Pc. Jumper Wire Kit, 5 cm Long

6. IoT Based Patient Health Monitoring System using ESP8266 and Arduino

Health monitoring is the major problem in today's world. Due to lack of proper health monitoring, patient suffer from serious health issues. There are lots of IoT devices now days to monitor the health of patient over internet. Health experts are also taking advantage of these smart devices to keep an eye on their patients. With tons of new healthcare technology start-ups, IoT is rapidly revolutionizing the healthcare industry.

Here in this project, we will make an IoT based Health Monitoring System which records the patient heart beat rate and body temperature and also send an email/SMS alert whenever those readings goes beyond critical values. Pulse rate and body temperature readings are recorded over ThingSpeak and Google sheets so that patient health can be monitored from anywhere in the world over internet. A panic will also be attached so that patient can press it on emergency to send email/sms to their relatives [8].

Materials required

1. Arduino Uno and Programming Cable
2. ESP8266 Wi-Fi module
3. LM35 temperature sensor
4. Pulse rate sensor
5. Push button
6. 10k Resistor
7. Male-female wires
8. Breadboard.

7. ECG Graph Monitoring with AD8232 ECG Sensor & Arduino

Heart diseases are becoming a big issue for the last few decades and many people die because of certain health problems. Therefore, heart disease cannot be taken lightly. By analyzing or monitoring the ECG signal at the initial stage this disease can be prevented. So we present this project, i.e ECG Monitoring with AD8232 ECG Sensor & Arduino with ECG Graph.

The AD8232 is a neat little chip used to measure the electrical activity of the heart. This electrical activity can be charted as an ECG or Electrocardiogram. Electrocardiography is used to help diagnose various heart conditions.

So in this project, we will interface AD8232 ECG Sensor with Arduino and observe the ECG signal on a serial plotter or Processing IDE [9].

8. IoT IR Thermometer using MLX90614 & ESP8266 on Blynk

In this project, we will build our own IoT Based IR Thermometer using MLX90614 & ESP8266 & monitor the temperature on the Blynk Application. This DIY Infrared Thermometer is the low-cost contactless Thermometer for measuring body temperature or the temperature of very hot bodies. An infrared thermometer is a device that measures the infrared radiation emitted by an object. Infrared radiation means a type of electromagnetic radiation below the visible spectrum of light.

The IR Temperature Sensor MLX90614 from Melexis can be easily connected to any microcontroller using the SMBus Protocol, which is similar to I2C Protocol. Here, we will interface MLX90614 Sensor with NodeMCU ESP8266 & 0.96" OLED Display [10].

9. IoT ECG Monitoring with AD8232 ECG Sensor & ESP8266

Heart diseases are becoming a big issue for the last few decades and many people die because of certain health problems. Therefore, heart disease cannot be taken lightly. So there should be a technology that can monitor the heart rate and heart behavior of the patient regularly. By analyzing or monitoring the ECG signal at the initial stage the various heart disease can be prevented.

This is the reason why I am presenting you with this great IoT project. In this project, I will show you how you can interface AD8232 ECG Sensor with NodeMCU ESP8266 Board and monitor the ECG Waveform on Serial Plotter Screen. Similarly, you can send the ECG waveform over the IoT Cloud platform and monitor the signal online from any part of the world using the PC or simply using the Smartphone. There is no need for staying in the Hospital to monitor heart activity/behavior just because you can monitor it online from anywhere. Thus it can be said advancement in Patient Health Monitoring System.

The IoT platform that I am gonna use here is Ubidots. Ubidots is an IoT Platform empowering innovators and industries to prototype and scale IoT projects to production. Use the Ubidots platform to send data to the cloud from any Internet-enabled device. You can then configure actions and alerts based on your real-time data and unlock the value of your data through visual tools [11].

10. Design of a Remote Real-Time Monitoring System for Multiple Physiological Parameters Based on Smartphone

Utilization of the widely used wearable sensor and smartphone technology for remote monitoring represents a healthcare breakthrough. This study aims to design a remote real-time monitoring system for multiple physiological parameters (electrocardiogram, heart rate, respiratory rate, blood oxygen saturation, and temperature) based on smartphones, considering high performance, autoalarm generation, warning transmission, and security through more than one method. Methods. Data on monitoring parameters were acquired by the integrated circuits of wearable sensors and collected by an Arduino Mega 250 R3. The collected data were transmitted via a Wi-Fi interface to a smartphone. A patient application was developed to analyze, process, and display the data in numerical and graphical forms. The abnormality threshold values of parameters were identified and analyzed to generate an autoalarm in the system and transmitted with data to a doctor application via a third-generation (3G) mobile network and Wi-Fi. The performance of the proposed system was verified and evaluated. The proposed system was designed to meet main (sensing, processing, displaying, real-time transmission, autoalarm generation, and threshold value identification) and auxiliary requirements (compatibility, comfort, low power consumption and cost, small size, and suitability for ambulatory applications). Results. System performance is reliable, with a sufficient average accuracy measurement (99.26%). The system demonstrates an average time delay of 14 s in transmitting data to a doctor application via Wi-Fi compared with an average time of 68 s via a 3G mobile network. The proposed system achieves low power consumption against time (4 h 21 m 30 s) and the main and auxiliary requirements for remotely monitoring multiple parameters simultaneously with secure data. Conclusions. The proposed system can offer economic benefits for remotely monitoring patients living alone or in rural areas, thereby improving medical services, if manufactured in large quantities [12].

11. Remote Wi-Fi DHT11 Temperature and Humidity I2C 2 X 16 LCD Display With Two ESP8266 and Visuino

ESP8266 modules are great lowcost standalone controllers with built in Wi-Fi, and I already made a simple Blink instructable with ESP8266 NodeMCU module.

The advantage of the ESP8266 over Arduino and other controllers is the builtin Wi-Fi. In this Instructable I will show you how with the help of Visuino you can use two ESP8266 modules to make a remote LCD Display for Temperature and Humidity DHT11 sensor.

In the Instructable, I will use 2 NodeMCU modules. One version 0.9, and the other 1.0. The NodeMCU are the easiest way to program and experiment with ESP8266 controllers. This Instructable however can easily be done with other modules, and the Sensor module can even use ESP-01 module as it needs only one GPIO pin to connect to the DHT11 sensor 2 NodeMCU ESP8266 boards (I used both NodeMCU 0.9, and NodeMCU 1.0 versions, but any other combination, or even stand alone ESP-12 will work) One DHT11 Sensor module I got from this cheap 37 sensors set One I2C 16x2 LCD Display (Back side of the LCD with the I2C adapter showed on Picture 2) 7 Female-Female jumper wires [13].

12. Advantage of this project

1. IOT Monitoring proves really helpful when we need to monitor & record and keep track of changes in the health parameters of the patient over a period of time. So with the IOT health monitoring, we can have the database of these changes in the health parameters. Doctors can take the reference of these changes or the history of the patient while suggesting the treatment or the medicines to the patient.
2. Hospital stays are minimized due to Remote Patient Monitoring.
3. Hospital visits for normal routine checkups are minimized.
4. Patient health parameter data is stored over the cloud. So it is more beneficial than maintaining the records on printed papers kept in the files. Or even the digital records which are kept in a particular computer or laptop or memory device like a pen- drive. Because there are chances that these devices can get corrupt and data might be lost. Whereas, in the case of IOT, cloud storage is more reliable and does have minimal chances of data loss [14].

13. Technological components

The diverse applications of RPM lead to numerous variations of RPM technology architecture. However, most RPM technologies follow a general architecture that consists of four components.: [15]

1. Sensors on a device that is enabled by wireless communications to measure physiological parameters.
2. DHT11 (humidity and temperature sensor).
3. kAD8232 ECG Sensor Module
4. MLX90614 (Infrared thermometer).
5. Sensors can connect back to a central database by WiFi or cellular communication protocols depending on the manufacturer.
6. Local data storage at patients' site that interfaces between sensors and other centralized data repository and/or healthcare providers.
7. Centralized repository to store data sent from sensors, local data storage, diagnostic applications, and/or healthcare providers.
8. Diagnostic application software that develops treatment recommendations and intervention alerts based on the analysis of collected data.

Depending on the disease and the parameters that are monitored, different combinations of sensors, storage, and applications may be deployed [15][16].

14. Architectural Design

In this paper, the patients' monitoring patient based-home system consist of hardware and software design. The hardware part deals with the mechanical and construction design, electrical, electronic circuitry, and analysis of ECG signal if it's normal or abnormal according to heart rate. The software part deals with the programming of Arduino and Ubidots.

15. Arduino board

The Arduino is connected to the three sensors namely DHT21 (humidity and temperature sensor), kAD8232 ECG Sensor Module, and MLX90614 (Infrared thermometer).

16. Implementation

The project is divided into two parts. First the hardware design and second the software design.

The project activity must be done step by step. It begins with searching and collecting information from a scientific perspective. The review is about mechanical structure design and electronic circuit design also software programming implementation. The project was begun by finding the concept and idea related to this title and in the end, the signal is sent to the doctor via e-mail so that he can diagnose the patient's condition.

17. 13 Benefits of Remote Patient Monitoring for Patients, Providers, Payers, and Healthcare Organizations [17]:

1. Improves data driven clinical decision making.
2. Helps patients improve self management and care plan adherence.
3. Cost of Care reduction for payers and providers.
4. Boosts net patient revenue.
5. Reduces patient's expenses and improves work productivity.
6. Improves access to care.
7. Builds patient engagement.
8. Optimizes clinical staff efficiency and combats clinical staff shortages.
9. Prevents the spread of infectious diseases and Hospital-Acquired Infections.
10. Boosts caregiver connectivity and involvement in care.
11. Improves patient experience and satisfaction.
12. Expands referral opportunities and improves retention.
13. Improves the clinician-patient relationship.

Chapter Three: System Requirements

1. Hardware components
 1. ESP8266.
 2. DHT21 (humidity and temperature sensor).
 3. kAD8232 ECG Sensor Module
 4. MLX90614 (Infrared thermometer).
 5. Bread board.
 6. Jumper wires.
 7. LCD 20×4

2 ESP8266

The ESP8266 is a low-cost Wi-Fi microchip, with built-in TCP/IP networking software, and microcontroller capability, produced by Espressif Systems [18] in Shanghai, China.

The chip first came to the attention of Western makers in August 2014 with the ESP-01 module, made by a third-party manufacturer Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first, there was almost no English-language documentation on the chip and the commands it accepted [19]. The very low price and the fact that there were very few external components on the module, which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, the chip, and the software on it, as well as to translate the Chinese documentation [20].

The ESP8285 is a similar chip with a built-in 1 MiB flash memory, allowing the design of single-chip devices capable of connecting via Wi-Fi [21].

➤ Nodemcu Versions or Generations:

As of now, there are three official versions of the Nodemcu: Nodemcu Devkit 0.9 or Nodemcu V1, Nodemcu Devkit 1.0 or Nodemcu V2, and Lolin Nodemcu or Nodemcu V3.

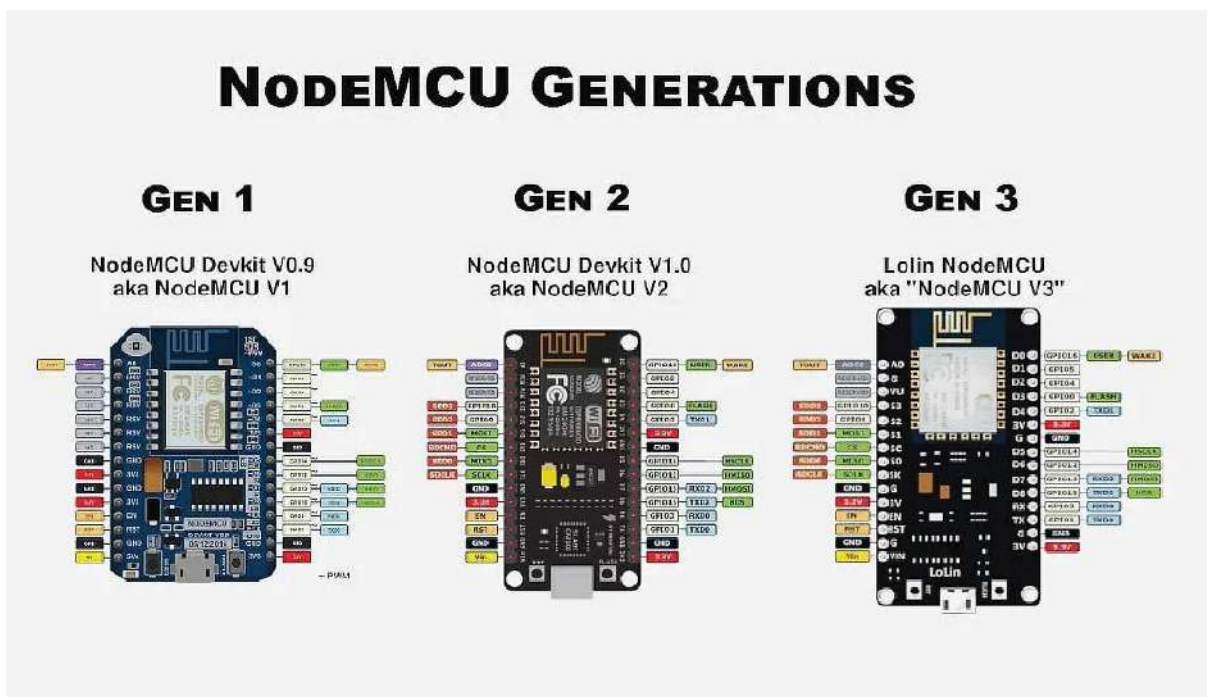


Figure 1 nodemcu esp8266 generations.

➤ Nodemcu Esp8266 packaging style:

There are two main packaging types of Nodemcu that are available in the market:

➤ Amica Nodemcu package:

The Amica Nodemcu has narrow pin spacing and measures 49mm x 26mm with a standard pin space of 0.1" between pins and 0.9" between rows.

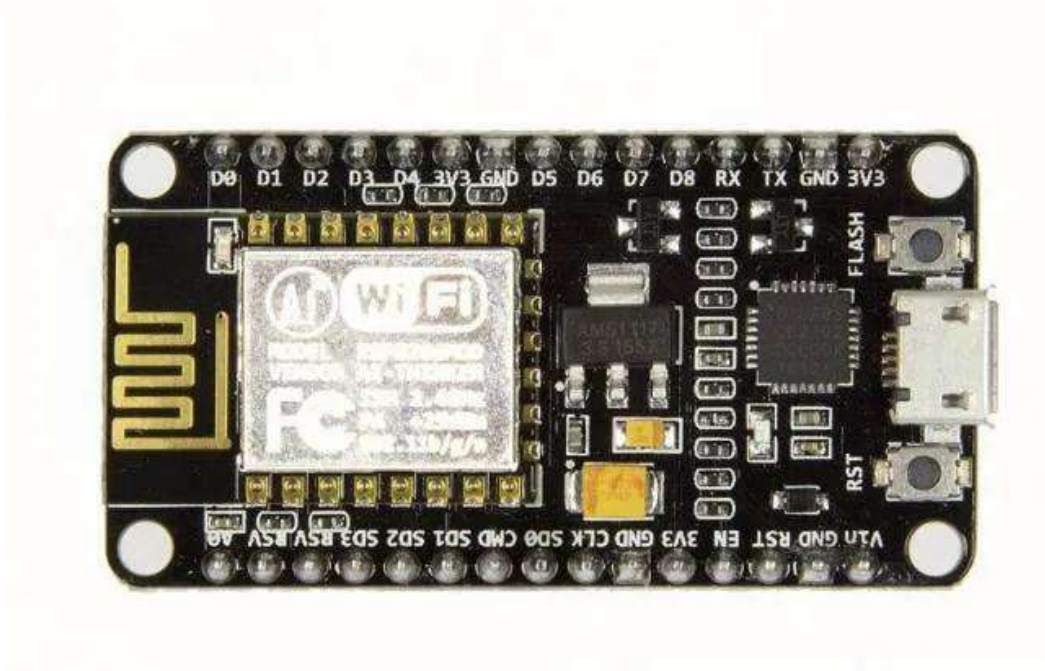


Figure 2 Nodemcu esp8266 board in Amica package.

➤ LoLin Nodemcu package:

The LoLin Nodemcu has a wider pin spacing and larger board layout than the Amica package. It measures 58mm x 32mm with a pin spacing of 0.1" between pins and 1.1" between rows.

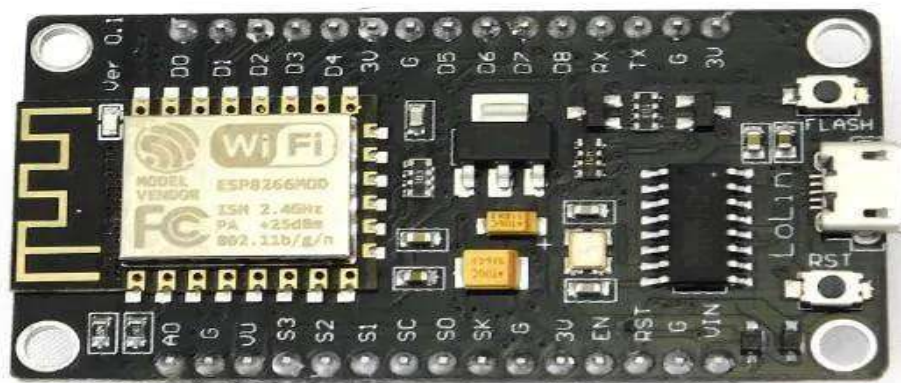


Figure 3 LoLin Nodemcu ESP8266 Board [24].

- Features of esp8266 :
- Processor: L106 32-bit RISC microprocessor core based on the Tensilica Xtensa Diamond Standard 106Micro running at 80 MHz [22].
- Memory:
 - ✓ 32 KiB instruction RAM
 - ✓ 32 KiB instruction cache RAM
 - ✓ 80 KiB user-data RAM
 - ✓ 16 KiB ETS system-data RAM

- ✓ External QSPI flash: up to 16 MiB is supported (512 KiB to 4 MiB typically included)
- ✓ IEEE 802.11 b/g/n Wi-Fi
- ✓ Integrated TR switch, balun, LNA, power amplifier and matching network
- ✓ WEP or WPA/WPA2 authentication, or open networks
- ✓ 17 GPIO pins [22]
- ✓ Serial Peripheral Interface Bus (SPI)
- ✓ I²C (software implementation) [23]
- ✓ I²S interfaces with DMA (sharing pins with GPIO)
- ✓ UART on dedicated pins, plus a transmit-only UART can be enabled on GPIO2
- ✓ 10-bit ADC (successive approximation ADC) [25]
- ESP8266 wifi module Specification :
 - ✓ 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2).
 - ✓ General-purpose input/output (16 GPIO).
 - ✓ Inter-Integrated Circuit (I²C) serial communication protocol.
 - ✓ Analog-to-digital conversion (10-bit ADC).
 - ✓ Serial Peripheral Interface (SPI) serial communication protocol.
 - ✓ I²S (Inter-IC Sound) interfaces with DMA(Direct Memory Access) (sharing pins with GPIO).
 - UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2).
 - Pulse-width modulation (PWM).

It employs a 32-bit RISC CPU (reduced instruction set computer) based on the Tensilica Xtensa L106 running at 80 MHz (or overclocked to 160 MHz). also It has a 64 KB boot ROM, 64 KB instruction RAM, and 96 KB data RAM. External flash memory can be accessed through SPI.

Table nodemcu ESP8266 Wifi Module Specification

| | |
|---------------------------|-----------------|
| Microcontroller | ESP-8266 32-bit |
| NodeMCU Model | Amica(official) |
| Clock Speed | 80-160 MHz |
| USB to Serial | CP2102 |
| USB Connector | Micro USB |
| Operating Voltage | 3.3V |
| Input Voltage | 4.5V-10V |
| Flash Memory/SRAM | 4 MB / 128 KB |
| GPIO Pins | 17 |
| Digital I/O Pins | 11 |
| Analog In Pins | 1 |
| PWM Pins | 4 |
| ADC Range | 0-3.3V |
| UART/SPI/I ² C | 2 / 1 / 1 |
| WiFi Built-In | 802.11 b/g/n |
| Temperature Range | -40C – 125C |

- power of ESP8266 :

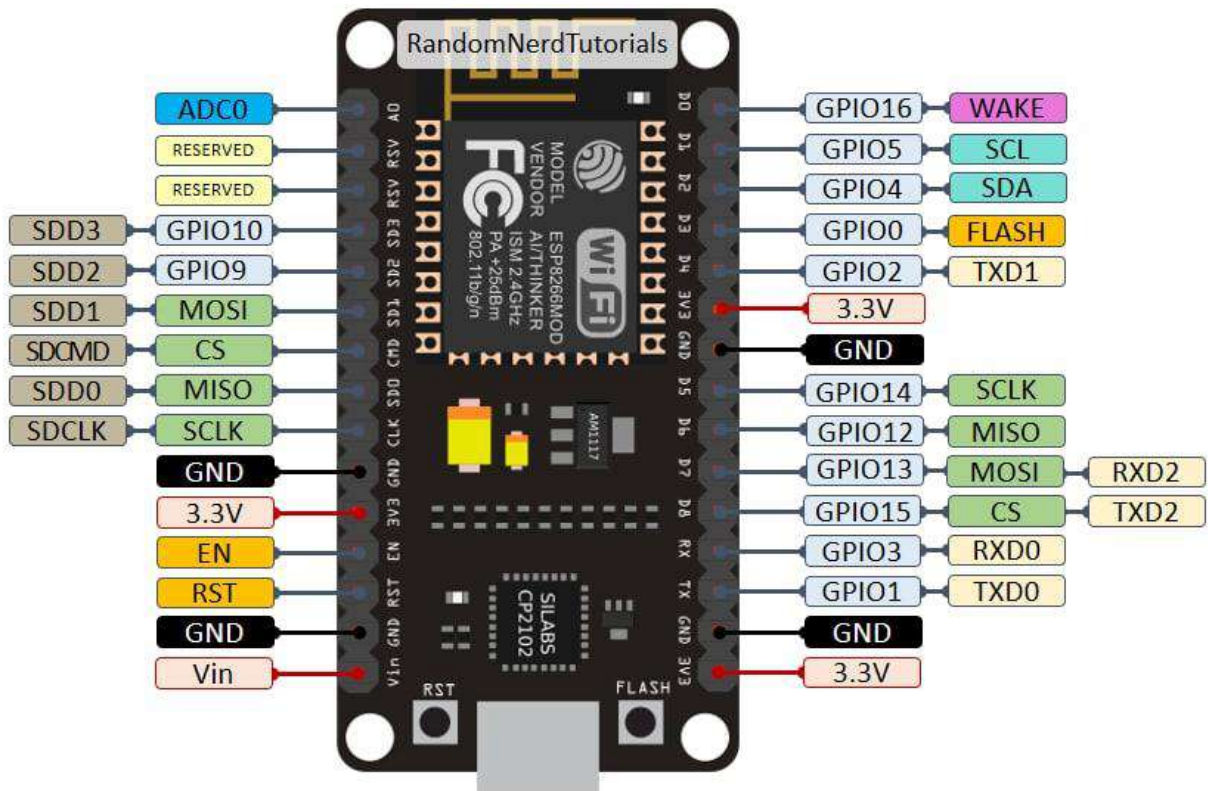
From the datasheet, we find nominal voltage of the esp8266 is 3.3V. Minimum voltage for stable operation is 2.5V and the maximum allowable limit is 3.6V. So for safely running an ESP8266

device, the battery voltage should be within 2.5 to 3.6V. An esp8266 chip can draw maximum 170mA current [26].

➤ Pinout :

From the previous chapter, you know that there is a great variety of boards and modules. So if you want to make a single switch, you will not use ESP-12F with a lot of GPIO. You'll probably use a small module like ESP-01.

The important thing is the marking of the pins. Unfortunately, every producer has his vision on how to do it. Therefore, the markings may (and in most cases will) vary.



➤ Description of all GPIOs :

ESP8266EX has 17 GPIO pins, which you can assign to various functions. They can act as I2C, I2S, UART, PWM, and IR Remote Control, etc.

| PIN | Direction | Description |
|-------|----------------|--|
| GPIO0 | INPUT / OUTPUT | GPIO0 is used to select booting mode. Pull it to GND during startup for flashing mode, pull to VCC (or leave it open) for normal boot mode. In most boards (level 3), it is connected to a flash button. If you use a module (level 2), it is convenient to design a button that shorts it to ground. In normal operation mode, it can be used as a classic GPIO. |
| GPIO1 | OUTPUT | GPIO1 is used as a UART0 TX during flash programming. On your PCB, connect it with the pins for the UART-USB adapter. In normal operation mode, it can be used as an output GPIO. It is HIGH at boot. If you pull it to GND during startup – boot fails. |
| GPIO2 | INPUT / OUTPUT | GPIO2 can be used as a classic GPIO. It is often connected to the LED. It is HIGH at boot. If you pull it to GND during startup – boot fails. |
| GPIO3 | INPUT | GPIO3 is used as a UART0 RX during flash programming. On your PCB, connect it with the pins for the UART-USB adapter. In normal operation mode, it can be used as an input GPIO. |

| | | |
|--------|--------------------|--|
| | | It is HIGH at boot. If you pull it to GND during startup – boot fails. |
| GPIO4 | INPUT / OUTPUT | GPIO4 often used as SDA for I2C (any other pin can be used as well). In normal operation mode, it can be used as a classic GPIO. |
| GPIO5 | INPUT / OUTPUT | GPIO4 often used as SCL for I2C (any other pin can be used as well). In normal operation mode, it can be used as a classic GPIO. |
| GPIO6 | Not recommended | GPIO6 is used to connect with the clock pin of the SD card. |
| GPIO7 | Not recommended | GPIO7 is used to connect with data pin 0 of the SD card. |
| GPIO8 | Not recommended | GPIO8 is used to connect with data pin 1 of the SD card. |
| GPIO9 | Not recommended | GPIO9 is used to connect with data pin 2 of the SD card. |
| GPIO10 | Not recommended | GPIO10 is used to connect with data pin 3 of the SD card. |
| GPIO11 | Not recommended | GPIO11 is used to connect with the command pin of the SD card. |
| GPIO12 | INPUT / OUTPUT | GPIO12 can be used as a MISO pin for the HSPI (High Speed Parallel Interface) interface. It can be a classic GPIO if you don't use HSPI. |
| GPIO13 | INPUT / OUTPUT | GPIO13 can be used as a MOSI pin for the HSPI interface. It can be a classic GPIO if you don't use HSPI. |
| GPIO14 | INPUT / OUTPUT | GPIO14 can be used as a CLK pin for the HSPI interface. It can be a classic GPIO if you don't use HSPI. |
| GPIO15 | INPUT / OUTPUT | GPIO15 can be used as a CS pin for the HSPI interface. It can be a classic GPIO if you don't use HSPI. It is LOW at boot. If you pull it to VCC during startup – boot fails. |
| GPIO16 | INPUT / OUTPUT | GPIO16 can be used to wake up from deep sleep, or as a classic GPIO. |
| ADC0 | INPUT | ADC0 (Analog to Digital Converter) is 10-bit precision analog input pin. |
| CH_EN | not applicable | Chip Enable pin: – if HIGH: chip works properly. – if LOW: small current consumed |
| RST | not applicable | External reset signal: – if HIGH: chip works properly. – if LOW: chip under reset |

1. Wi-Fi Key Features :

- 802.11 b/g/n support
- 802.11 n support (2.4 GHz), up to 72.2 Mbps
- Defragmentation
- 2 x virtual Wi-Fi interface
- Automatic beacon monitoring (hardware TSF)
- Support Infrastructure BSS Station mode/SoftAP mode/Promiscuous mode
- Applications :
- Home appliances
- Home automation

- Smart plugs and lights
- Industrial wireless control
- Baby monitors
- IP cameras
- Sensor networks
- Wearable electronics
- Wi-Fi location-aware devices
- Security ID tags
- Wi-Fi position system beacons [27]

2. kAD8232 ECG Sensor Module

The AD8232 ECG Module is a cost-effective board used to measure the electrical activity of the heart. This electrical activity can be charted as an ECG or Electrocardiogram and output as an analog reading. Additionally, this board includes pins like the right arm (RA), left arm (LA) & right leg (RL) pins to connect custom sensor

- Features of the AD8232 ECG Module :
 - ✓ Fully integrated single-lead ECG front end
 - ✓ Common-mode rejection ratio: 80 dB (dc to 60 Hz)
 - ✓ Two or three-electrode configurations
 - ✓ Qualified for automotive application
 - ✓ Single-supply operation: 2.0 V to 3.5
 - ✓ Fast restore feature improves filter settling
 - ✓ Size: 3.5cm x 3cm
- Application of AD8232 ECG Module :
 - ✓ Fitness and activity heart rate monitors
 - ✓ Portable ECG
 - ✓ Remote health monitors
 - ✓ Gaming peripherals
 - ✓ Biopotential signal acquisition

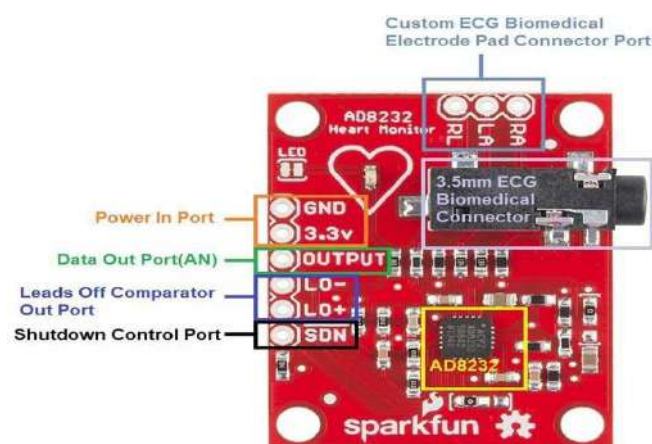


Figure 5 ECG sensor.

3. DHT21/AM2301 (humidity and temperature sensor)

The AM2301 / DHT21 temperature and humidity sensor consists of two sensors: A capacitive humidity sensor and a thermistor. This sensor transmits data using a single digital pin, so it can communicate with the microcontroller using only one digital pin. The sampling frequency is 0.5Hz, which means that you can get new data every two seconds [28].

3.1.3.1 DHT21/AM2301 Temperature and Humidity Sensor Pinout

This Module has 3 wires [28]:

- VCC (Red): Module power supply – 5V
- GND (Black): Ground
- DATA (Yellow): Data pin



Figure 6 pinout of DHT21/AM2301.

4. MLX90614 (Infrared thermometer)

The MLX90614 is an Infra-Red thermometer for non-contact temperature measurements. Both the IR sensitive thermopile detector chip and the signal conditioning ASSP are integrated in the same TO-39 can. Thanks to its low noise amplifier, 17-bit ADC and powerful DSP unit, a high accuracy and resolution of the thermometer is achieved. The thermometer comes factory calibrated with a digital PWM and SMBus (System Management Bus) output. As a standard, the 10-bit PWM is configured to continuously transmit the measured temperature in range of - 20...120°C, with an output resolution of 0.14°C. The factory default POR setting is SMBus [29].



Figure 7 Infrared thermometer

5. LCD 20x4 blue backlight 5V

The display is composed of a 20 character x 4 line LCD display with a blue backlight and white characters. Each of the characters are composed of a 5 x 8 dot matrix for good character representation. Custom characters can be defined and used with the display.

The backlight has a VO (Display Contrast) input for connecting a potentiometer for adjustment of the contrast of the display for best viewing. The potentiometer can be somewhere in the 10K-50K range and should connect between 5V and ground. The wiper output of the potentiometer feeds the VO pin with a variable voltage that sets the contrast. If this is not adjusted correctly the display may not show any characters or solid blocks may be displayed.

The backlight is powered off the A (Anode) and K (Cathode) pins. Normally 5V is applied to 'A' and ground to 'K'. The backlight includes a 51 ohm current limiting resistor, so an external resistor is not required.

The backlight can go down to about 3.2V before it goes out. It can be operated off the 3.3V line if a lower brightness is desired for the application. The backlight can be driven by a logic pin using a transistor if PWM brightness or ON/OFF control is desired [30].

➤ Technical Specifications

- ✓ Gross Weight (kg) 0.1000.
- ✓ Part Number 2004A.
- ✓ Display Format 20×4 Character.
- ✓ Interface 6800 4-bit Parallel , 6800 8-bit Parallel.
- ✓ IC or Equivalent AIP31066 , HD44780, KS0066 , SPLC780 , ST7066.
- ✓ Appearance White on Blue.
- ✓ Diagonal Size No.
- ✓ Connection Pin Header.
- ✓ Outline Dimension 98.00(W)x60.0(H)x14.0(T)mm.
- ✓ Visual Area 76.00×25.20mm.
- ✓ Active Area 70.40(W)x20.80(H)mm.
- ✓ Character Size 2.95×4.75mm.
- ✓ Dot (Pixel) Size 0.55×0.55mm.
- ✓ Dot (Pixel) Pitch 0.60×0.60mm.
- ✓ IC Package COB.
- ✓ Display Type STN-LCD Blue.
- ✓ Touch Panel Optional No.
- ✓ Sunlight Readable No.
- ✓ Response Time(Typ) No.
- ✓ Contrast Ratio(Typ) No.
- ✓ Colors No.
- ✓ Viewing Direction 6:00.
- ✓ Viewing Angle Range No.
- ✓ Brightness(Typ) No.

- ✓ Backlight Color White Color.
- ✓ Backlight Current (Typ) 75mA.
- ✓ Power Supply(Typ) 3.3V, 5V.
- ✓ Supply Current for LCM(Max) 2000uA.
- ✓ Operating Temperature -20°C~70°C.
- ✓ Storage Temperature -30°C~80°C [31].



Figure 8 LCD pins.

2. Software

3.2.1. Introduction to Arduino IDE

The most universal and easiest way to program any ESP8266 chip:

- Connect the USB-UART adapter to ESP8266 as follows: VCC -> VCC, GND -> GND, RX -> TX and TX -> RX.
- Pull the GPIO0 pin to GND.
- Connect the adapter to the computer.
- Run a program for flashing via UART, e.g. ESP Easy.
- Select the appropriate COM port and binary file you want to upload.
- Press the Flash button.
- After success, reset ESP8266.

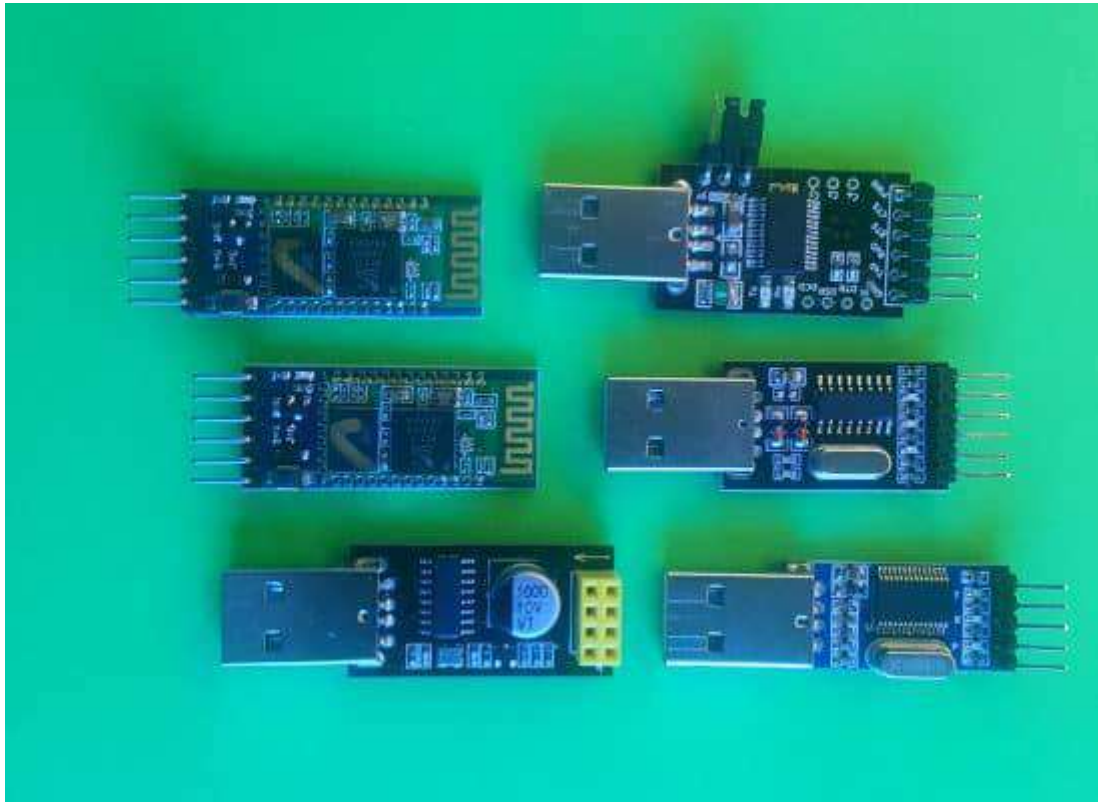


Figure 9 Connect ESP8266 to UART-USB adapter.

The above procedure is probably the most universal. You can program any ESP chips this way. It doesn't matter what you used to generate the binary file. Use an environment that you know and like the most.

Some of them, like the Arduino IDE, have a built-in programmer. So you don't have to use any other external program to flash the ESP8266 chip. Others like ESPHome can program OTA (Over the Air). We will follow these two ways below.

3. Arduino IDE Definition

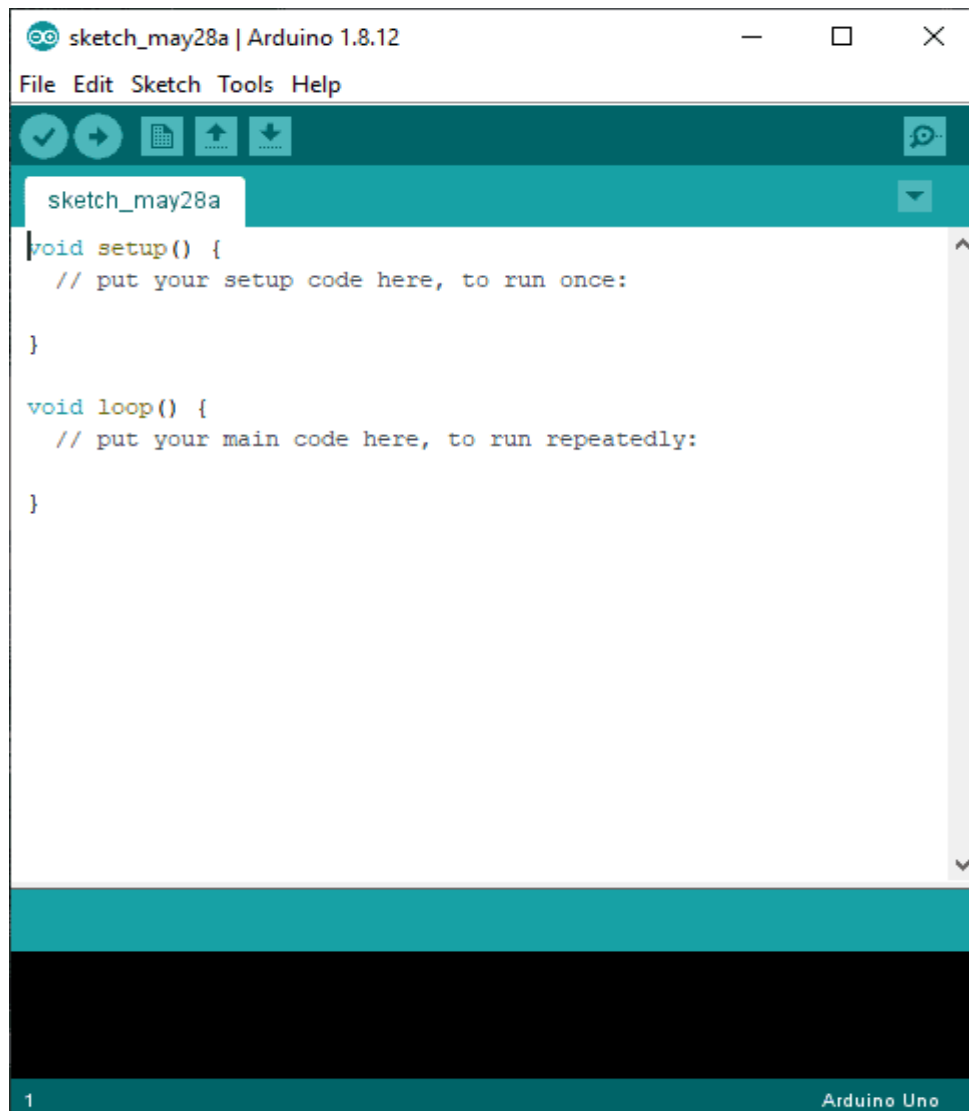
1. Arduino IDE is open-source software that is mainly used for writing and compiling the code into the esp8266 Module.
2. It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing, and compiling the code in the environment.
3. A range of esp8266 modules available including ESP8266 12-E NodeMCU, WeMos D1 Mini, ESP-01 and many more.
4. Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
5. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
6. The IDE environment mainly contains two basic parts: Editor and Compiler where the former is used for writing the required code and later is used for compiling and uploading the code into the given esp8266 Module.
7. This environment supports both C and C++ languages.

4. Programming the ESP8266 with the Arduino IDE

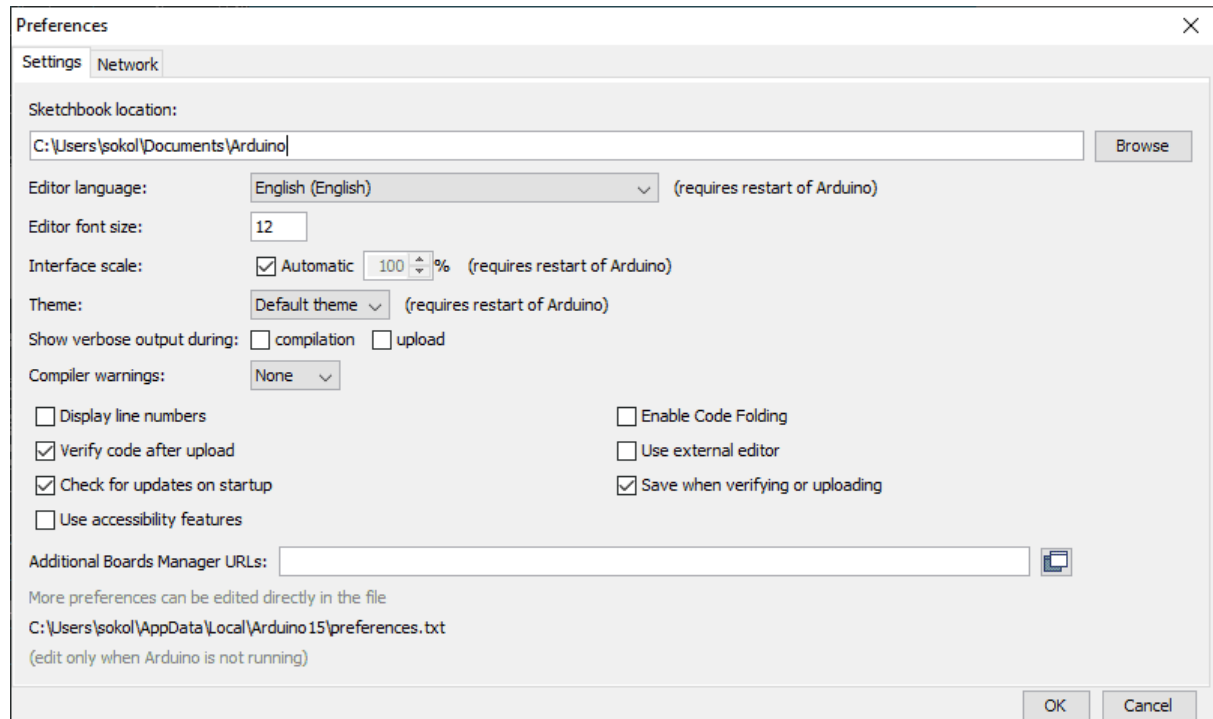
Esp8266 itself needs no introduction. It is an open-source hardware and software platform. And it is extremely popular among electronics hobbyists. No wonder, all their products are well polished and easy to use. But the biggest advantage is the community gathered around it. Thanks to them you can program the ESP8266 chip using the IDE.

The first step is connecting the UAR-USB adapter to the ESP chip. I described how to do it in the chapter Connect ESP8266 to UART-USB adapter. Remember to pull GPIO0 to GND before connecting the USB.

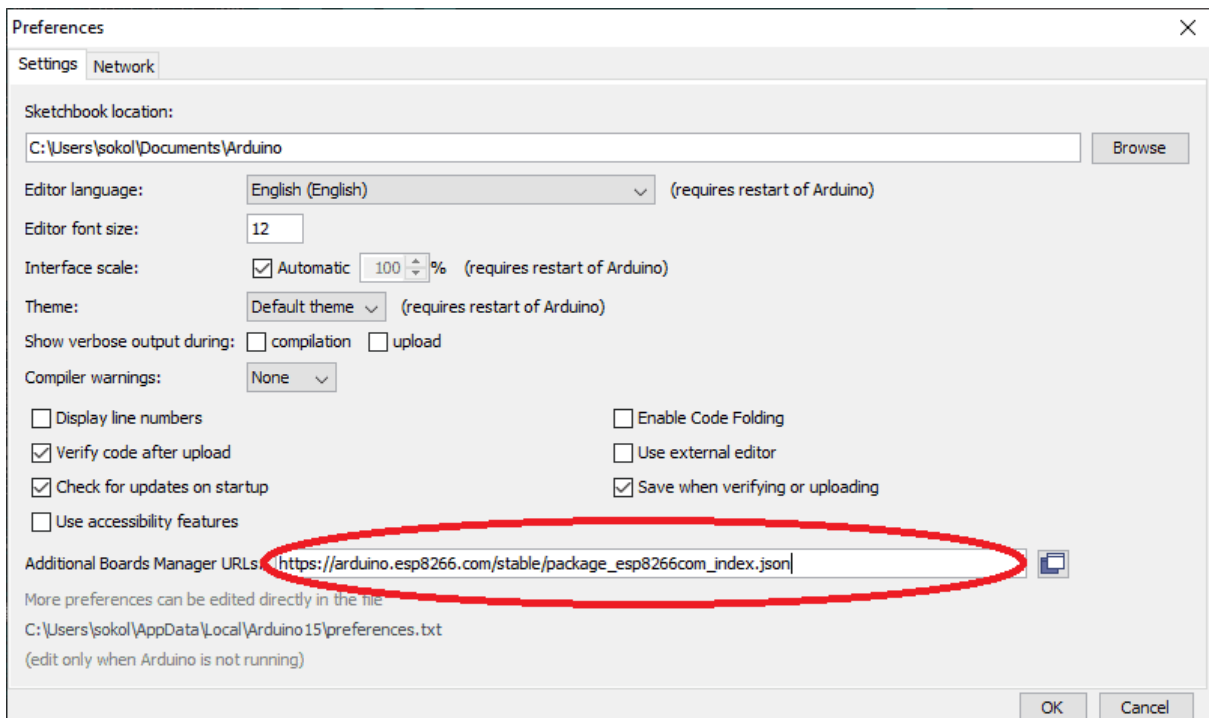
- Run Arduino IDE.



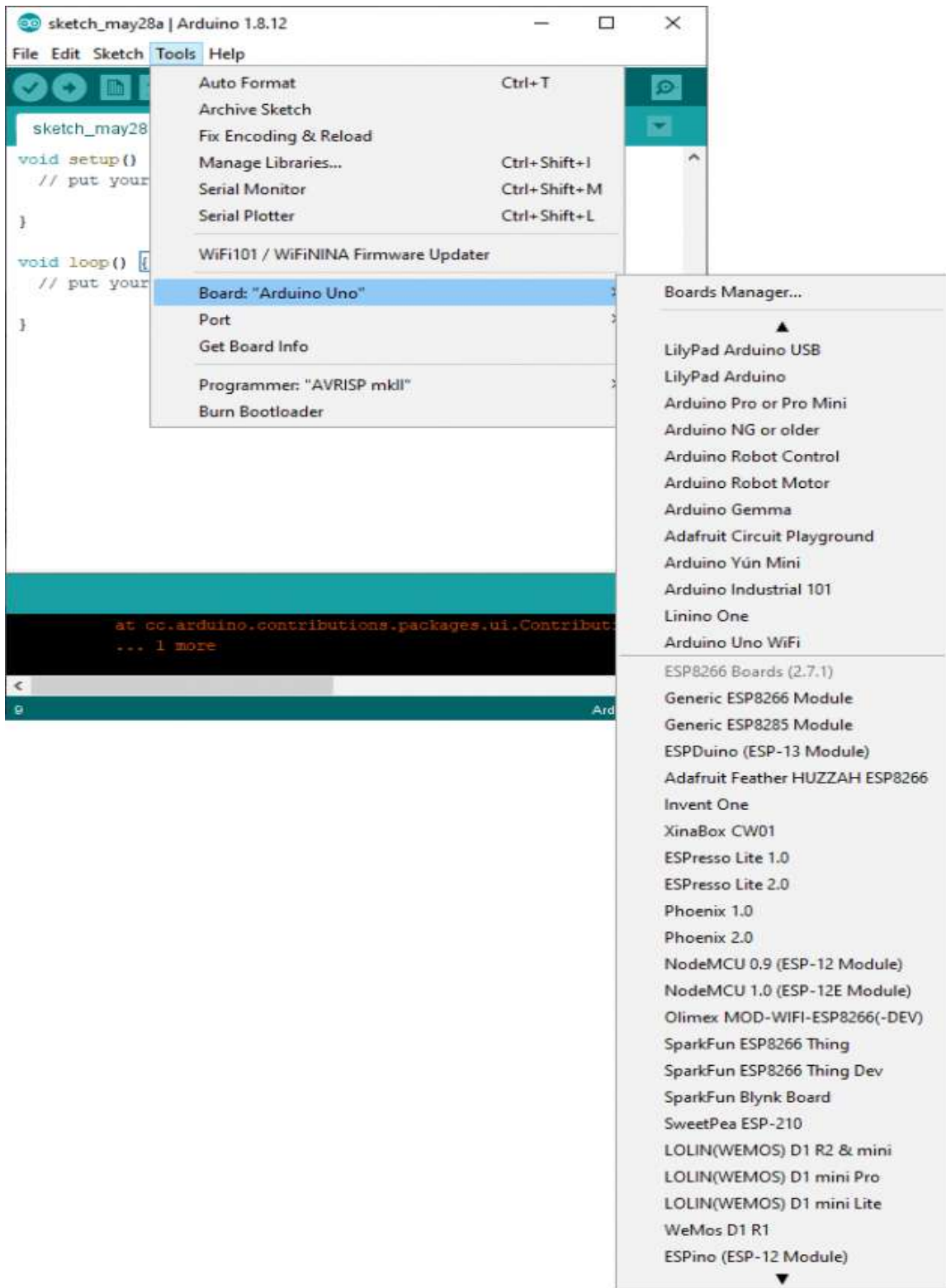
- Now we will just use the community work Add to IDE support for modules and boards with ESP8266. Go to File -> Preferences
- Enter: https://arduino.esp8266.com/stable/package_esp8266com_index.json into the Additional Board Manager URLs field. If you want to add more boards, just separate the URLs with a comma.



https://arduino.esp8266.com/stable/package_esp8266com_index.json into the Additional Board Manager URLs field. If you want to add more boards, just separate the URLs with a comma.

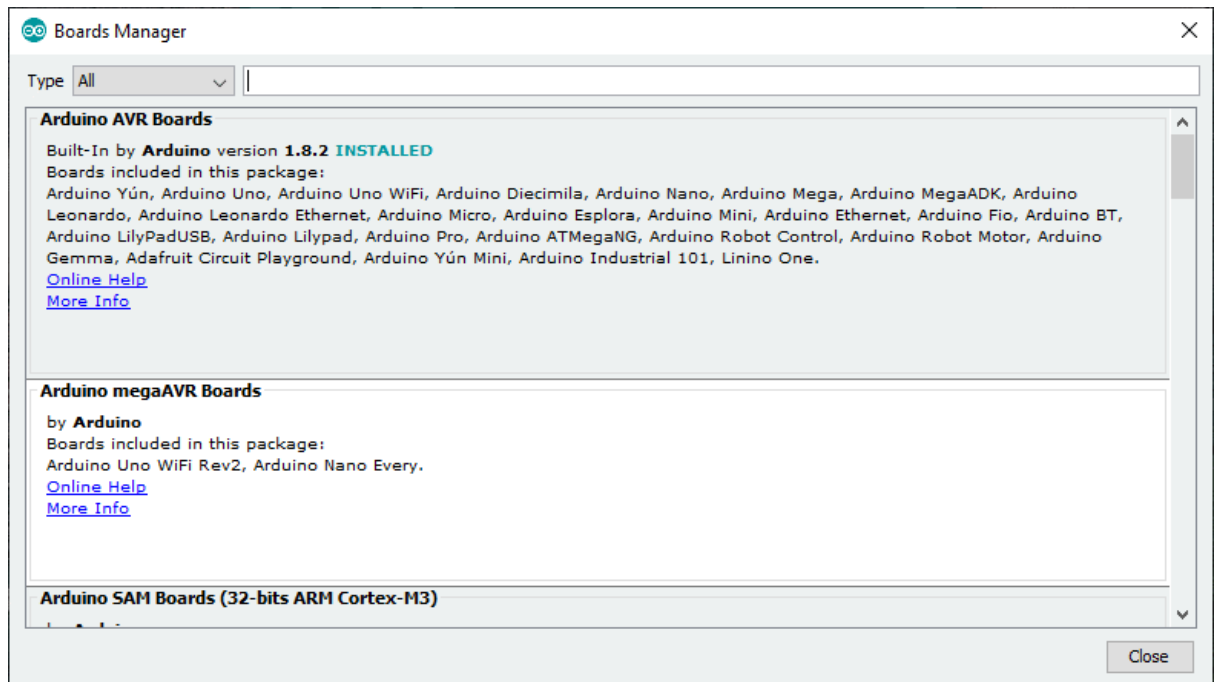


- Click OK. Now go to the Tools -> Board: "SOME NAME" -> Boards Manager...

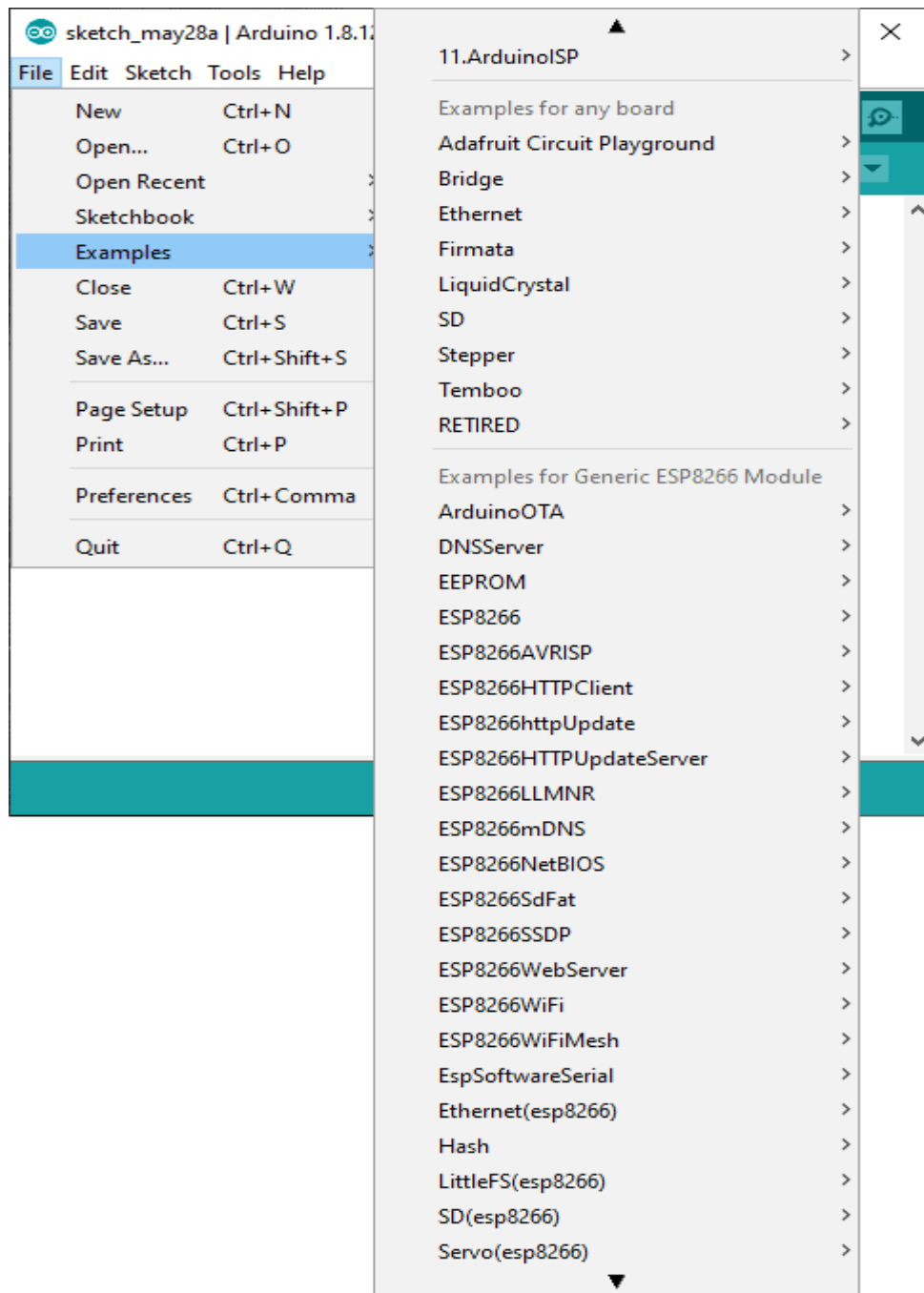


- Type “esp8266” in the field at the top or scroll all the way down. You will find there a package of libraries and examples for various ESP modules. Click Install

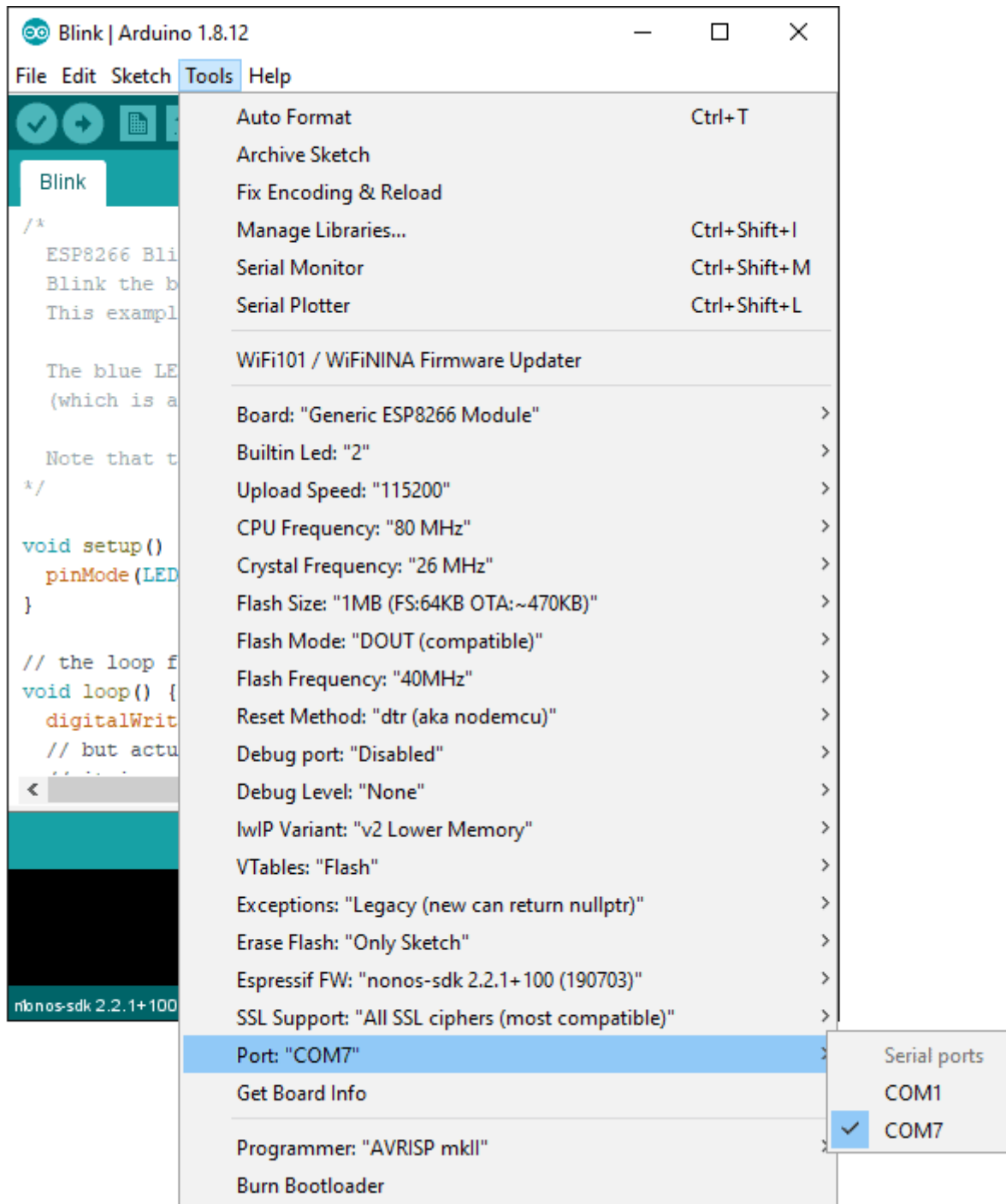
-
-



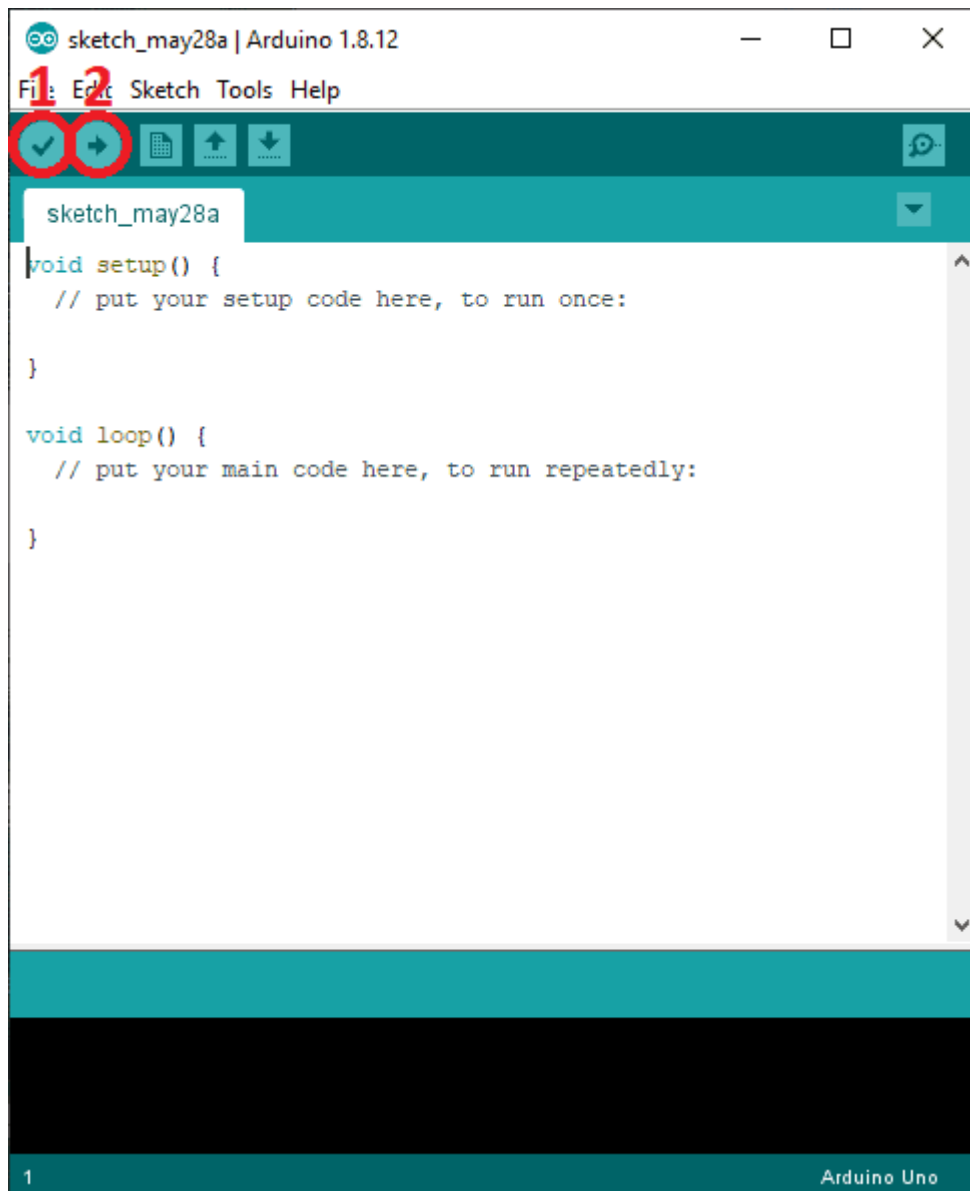
- Now you can go to Tools -> Board: "SOME NAME", and select the board you are interested in. E.g., Generic ESP8255 Module.
- Now you can write the code. But don't worry, we won't do it right now. At this point, you can use the ready sketch from anywhere. Often people share sketches on forums or Facebook groups. You can also use the examples you just installed.
- To do this go to File -> Examples. And choose the one you want. I recommend Blink to start with.



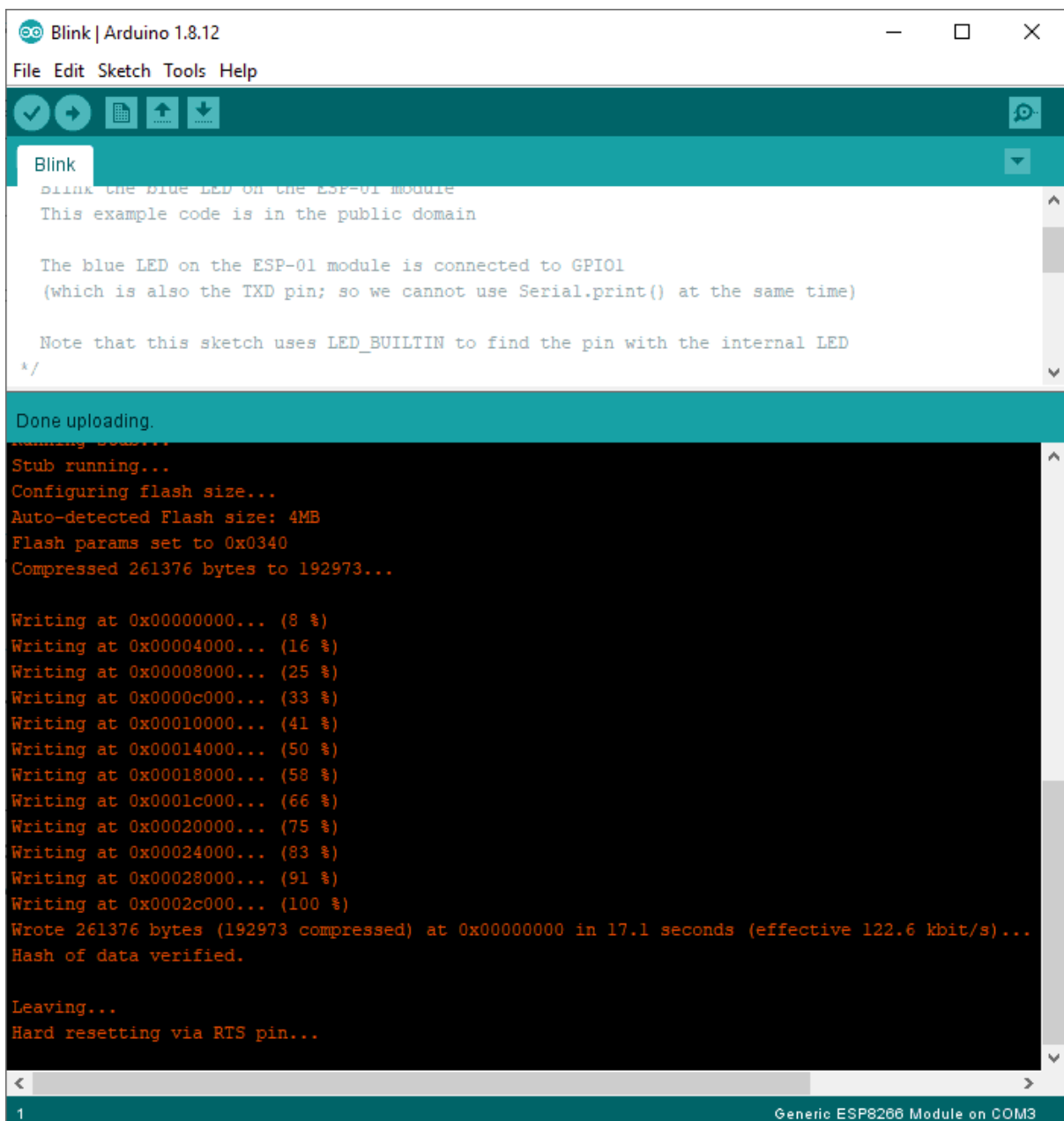
- I assume you've already connected your module to the computer via the UART-USB adapter. You must select the correct port number. This number is assigned by the OS when you connect the device (adapter) for the first time.



➤ There is only verification left and you can program chip ESP. Click Verify and then Upload.



➤ If everything went well, you should see the following window:



```

Blink | Arduino 1.8.12
File Edit Sketch Tools Help
Blink
Blink the blue LED on the ESP-01 module
This example code is in the public domain

The blue LED on the ESP-01 module is connected to GPIO1
(which is also the TXD pin; so we cannot use Serial.print() at the same time)

Note that this sketch uses LED_BUILTIN to find the pin with the internal LED
*/

Done uploading.
Stub running...
Configuring flash size...
Auto-detected Flash size: 4MB
Flash params set to 0x0340
Compressed 261376 bytes to 192973...

Writing at 0x00000000... (8 %)
Writing at 0x00004000... (16 %)
Writing at 0x00008000... (25 %)
Writing at 0x0000c000... (33 %)
Writing at 0x00010000... (41 %)
Writing at 0x00014000... (50 %)
Writing at 0x00018000... (58 %)
Writing at 0x0001c000... (66 %)
Writing at 0x00020000... (75 %)
Writing at 0x00024000... (83 %)
Writing at 0x00028000... (91 %)
Writing at 0x0002c000... (100 %)
Wrote 261376 bytes (192973 compressed) at 0x00000000 in 17.1 seconds (effective 122.6 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

1 Generic ESP8266 Module on COM3

```

Now all you have to do is reset the board and everything should work correctly [29].

3.2.4 Ubidots

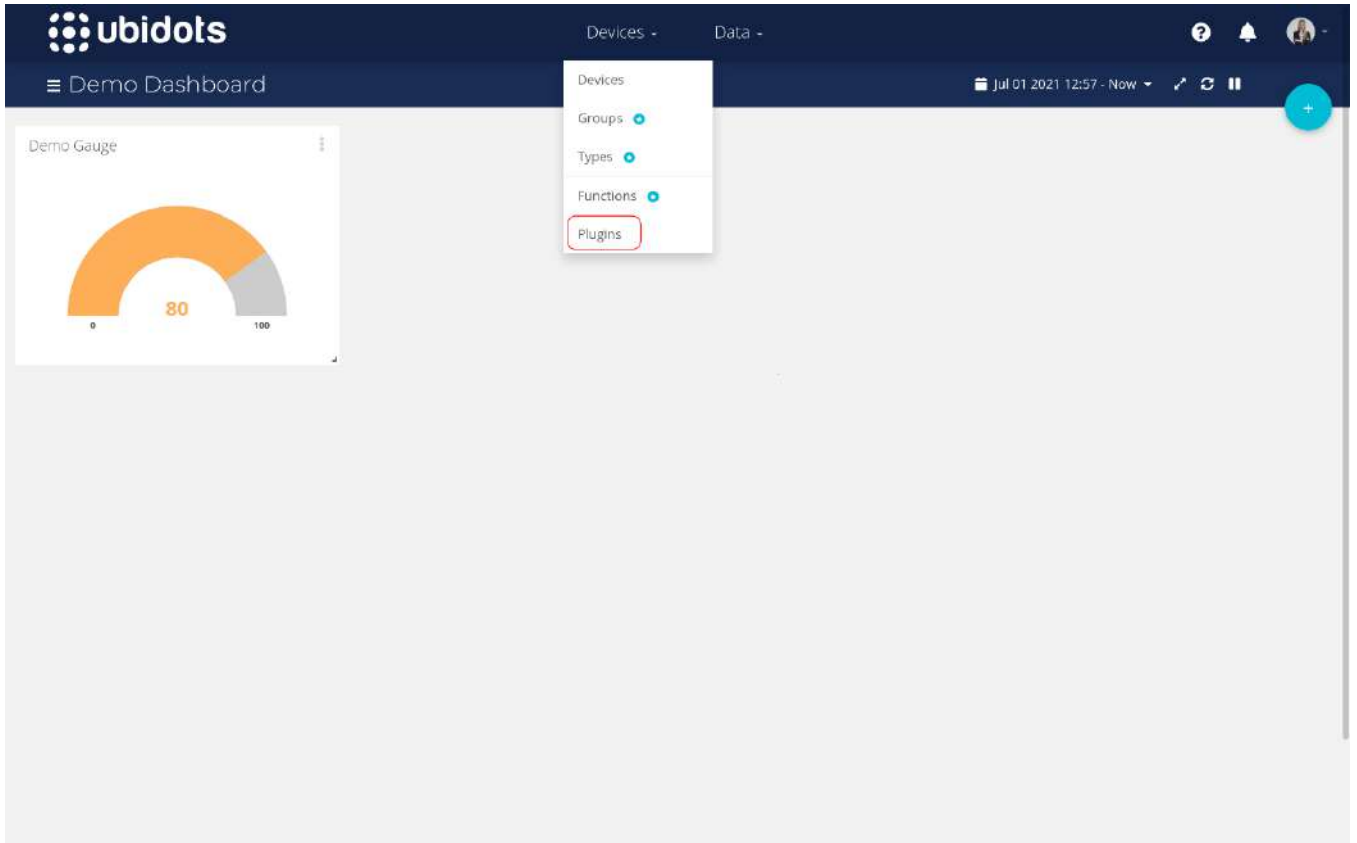
Ubidots provides a secure and easy way to build IoT solutions for students, makers, and researchers. It is used for sending data from any Internet-enabled device to the cloud, triggering actions and alerts based on that data, and visualizing it [32].

3.2.4.1 Prerequisites

A Ubidots user account.

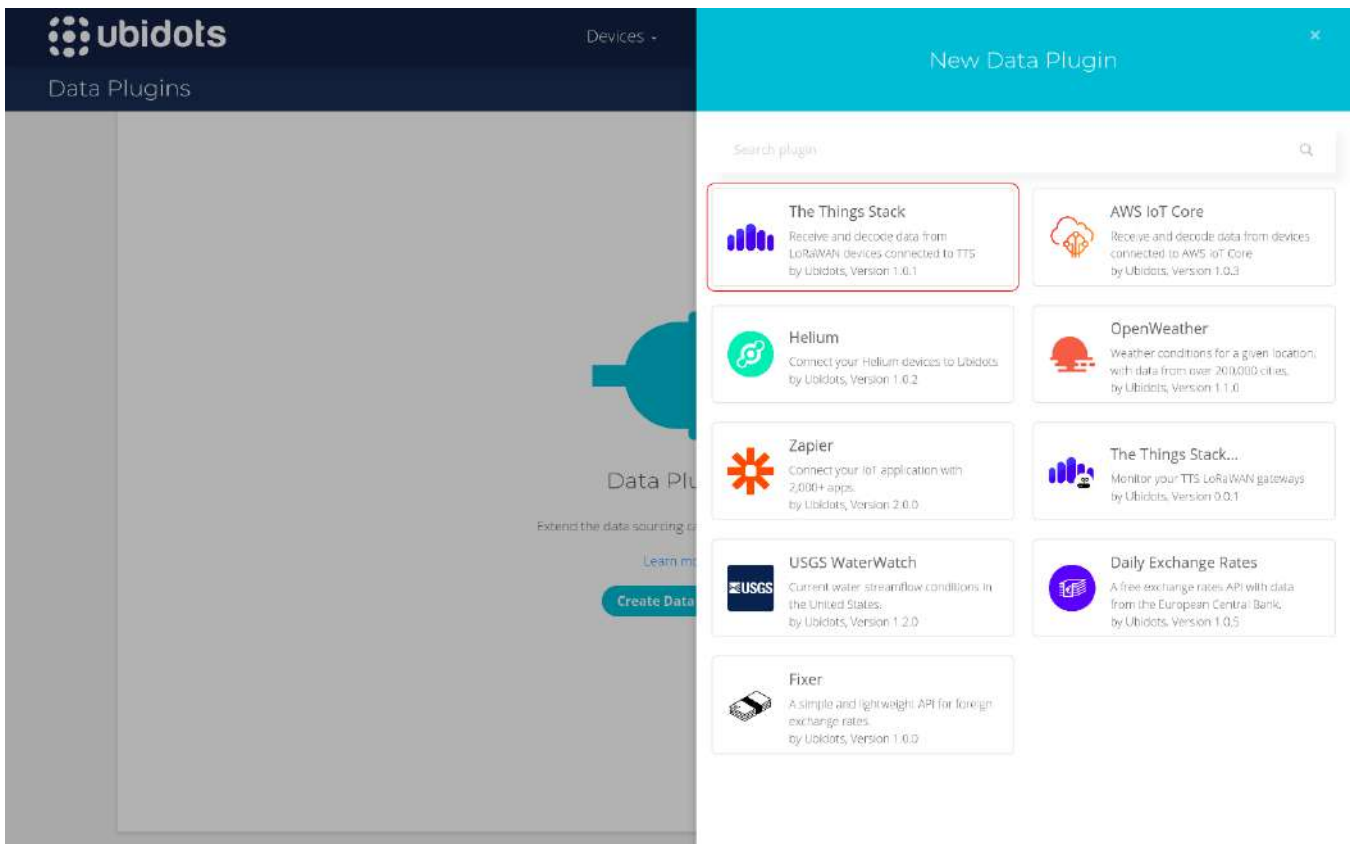
3.2.4.1.1 Setup Ubidots

Log in to your Ubidots account and find the Devices tab in the upper part of your dashboard. In its drop-down list, choose Plugins



Click on the + or on the Create Data Plugin button to create a new plugin.

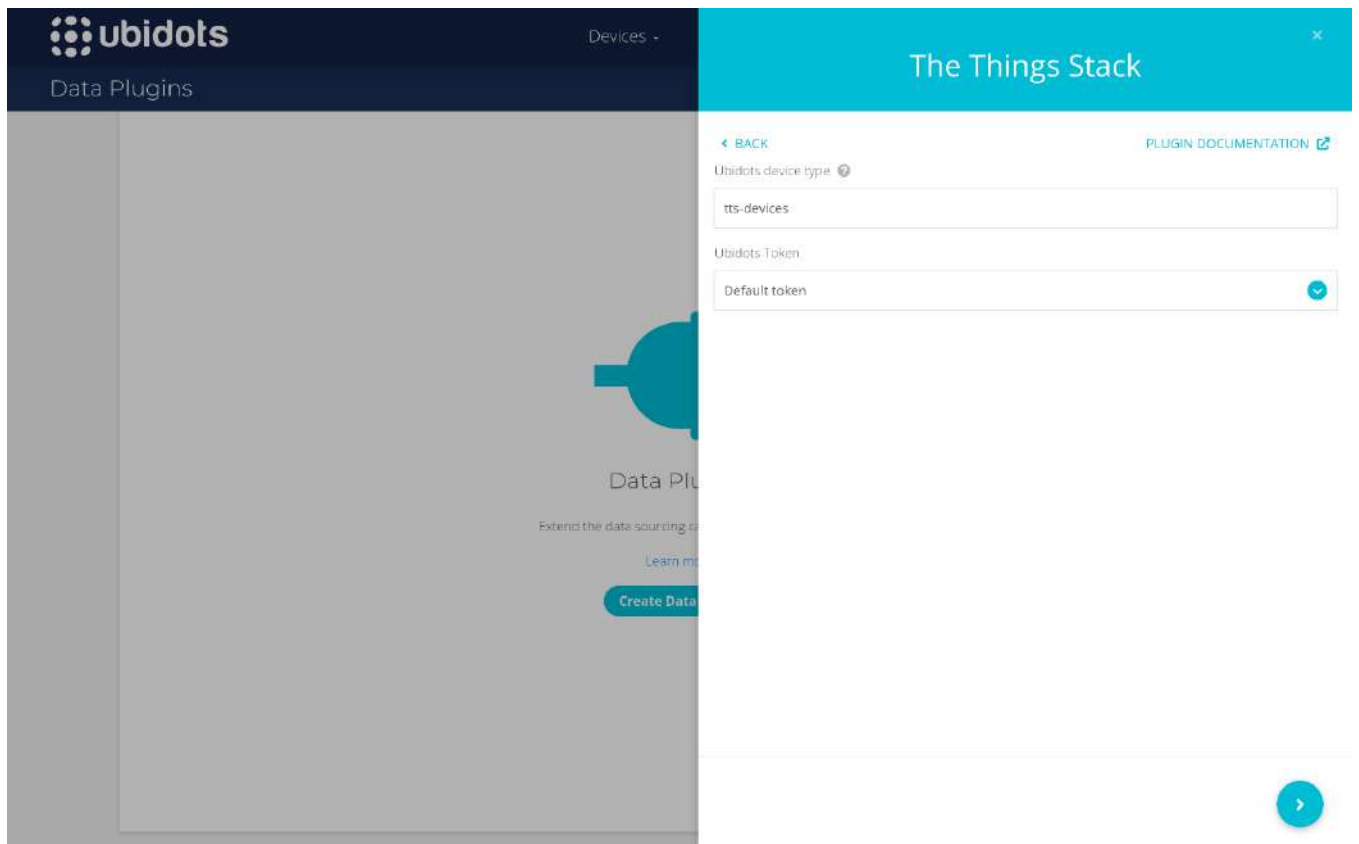
When you are presented with available plugins, select The Things Stack plugin.



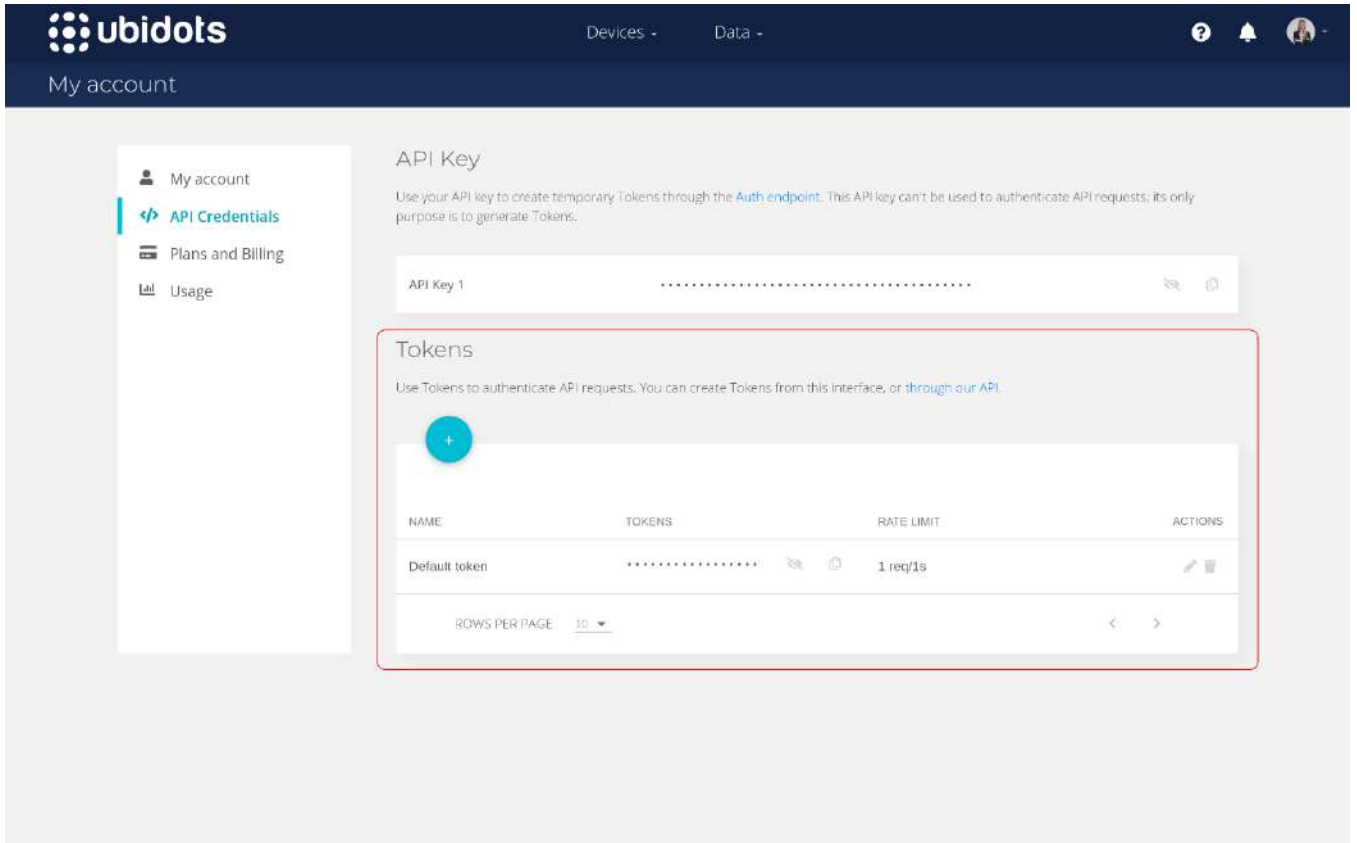
You will see details about Inputs, Usage, etc. Move forward by clicking the > button.

Next, you need to provide the name for a Ubidots device type after which a new device type will be created and linked to this plugin. This device type will later allow you to make changes for all devices that receive data through this plugin.

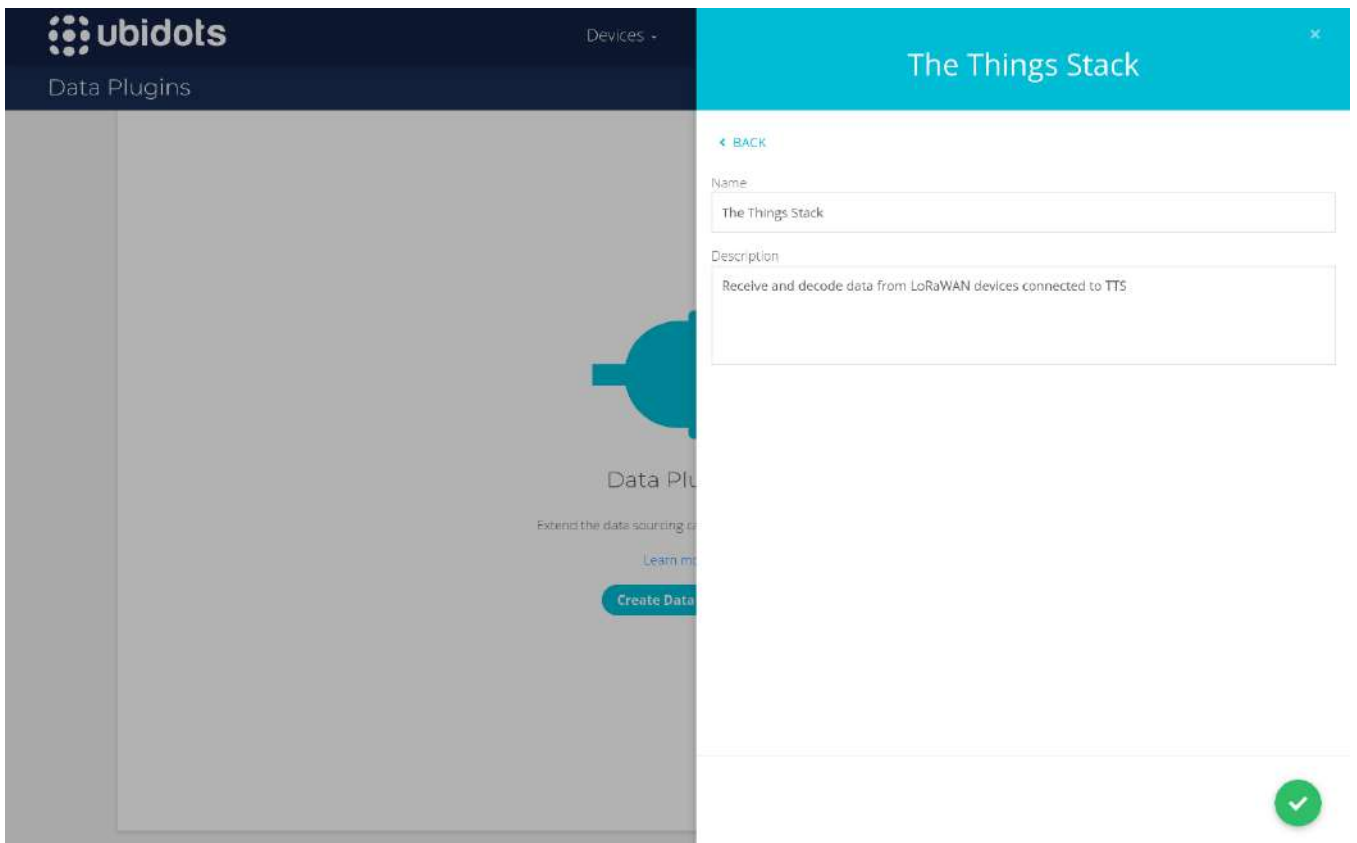
You also need to select a Ubidots token. You can use the Default token, but creating a new token specifically for this plugin is recommended.



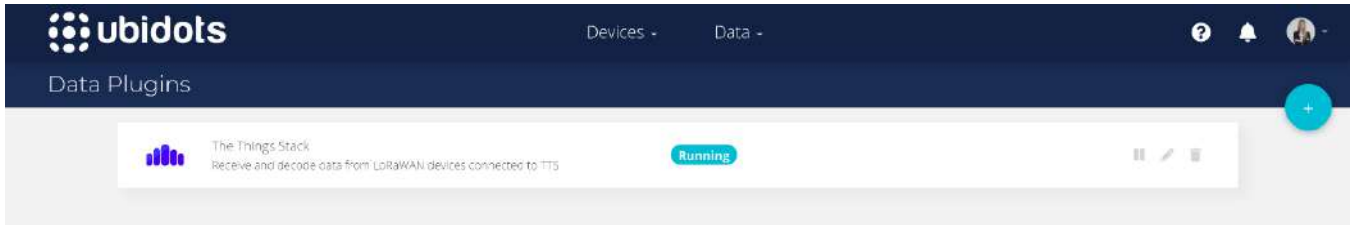
To create a new token, first, click on your avatar in the upper right corner and select API Credentials. Then select More below the Default token and add a new token within the API Credentials page.



Select > to continue and then hit the checkmark to finish.

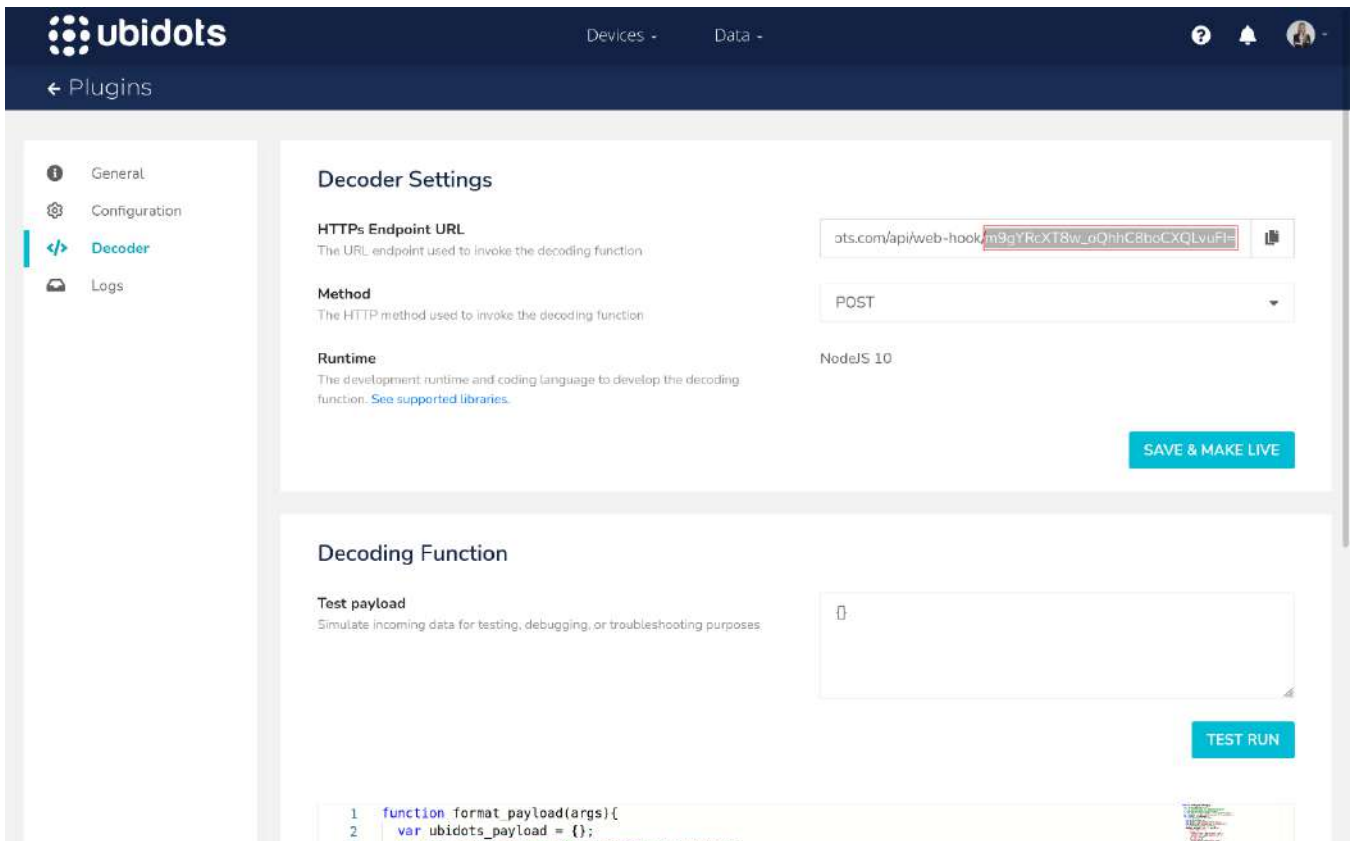


After a few moments, your newly created plugin will show the status Running.



To complete the integration with The Things Stack, you will need to provide the plugin ID and aforementioned token.

To find the plugin ID, click on your newly created plugin and navigate to the Decoder tab on the left. The plugin ID is available as part of the HTTPS Endpoint URL.



You can find Ubidots' pre-loaded sample payload decoder in your newly created plugin's menu, if you navigate to Decoder on the left and scroll down to the Decoding Function window. There you can edit it and finish with Save & Make Live.

The screenshot displays the Ubidots web interface. At the top, the Ubidots logo is on the left, and navigation links for 'Devices' and 'Data' are on the right. Below the logo, a 'Plugins' menu is visible. The main content area is divided into two sections:

- Decoder Settings:** This section includes:
 - HTTPs Endpoint URL:** A text input field containing 'https://datapugin.ubidots.com/api/web-hook/m9gYRcXT8w'.
 - Method:** A dropdown menu set to 'POST'.
 - Runtime:** A dropdown menu set to 'NodeJS 10'.
 - A 'SAVE & MAKE LIVE' button.
- Decoding Function:** This section includes:
 - Test payload:** A text area for simulating incoming data.
 - TEST RUN** button.
 - Code Editor:** A code editor showing a JavaScript function:


```
1 function format payload(args){
2   var ubidots_payload = {};
3   // Use received data for debugging purposes
```

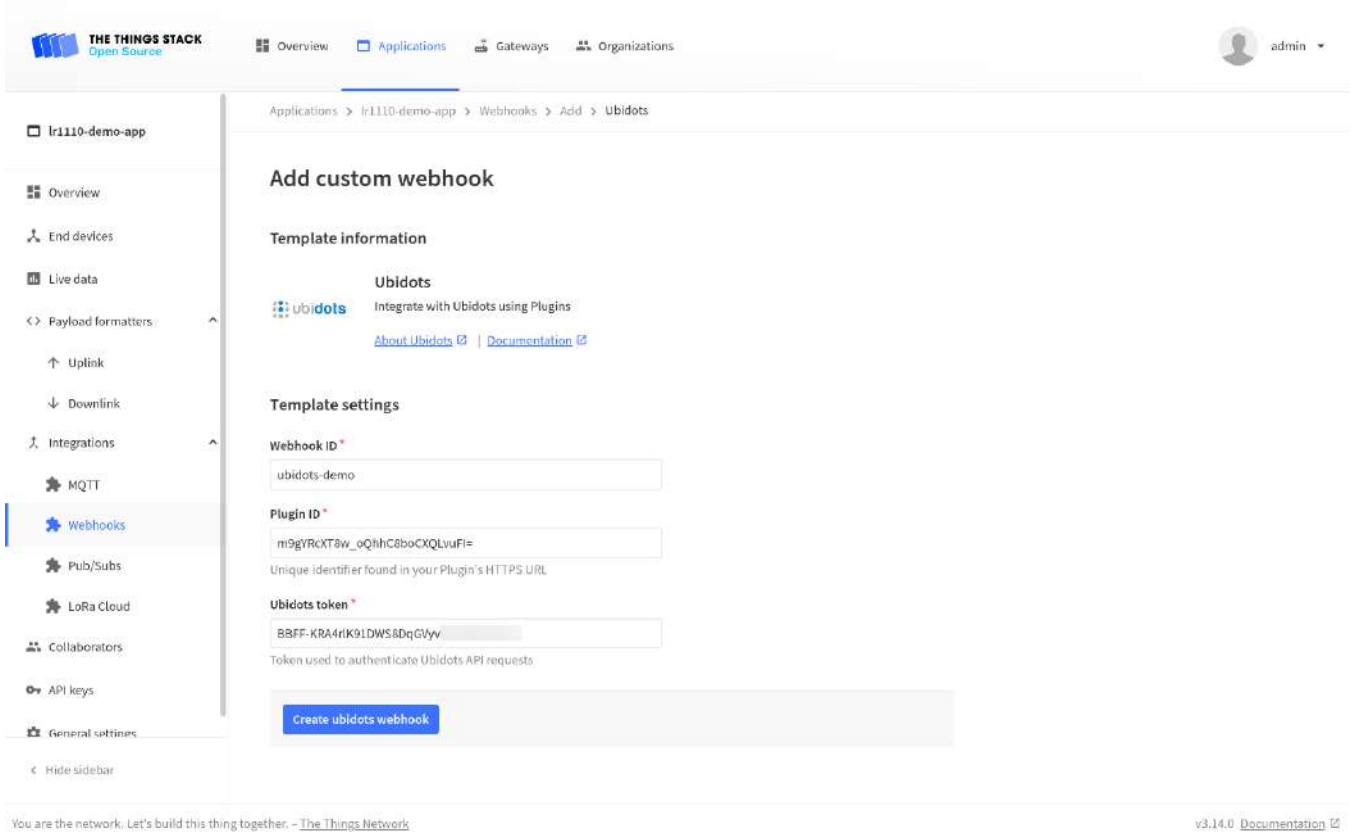
3.2.4.1.2 Configure the Things Stack

When you have prepared the setup on Ubidots, it is the time to create a Webhook integration on The Things Stack by using the Ubidots Webhook template.

On The Things Stack, navigate to Integrations → Webhooks and choose the Ubidots Webhook template.

Name your integration by filling in the Webhook ID.

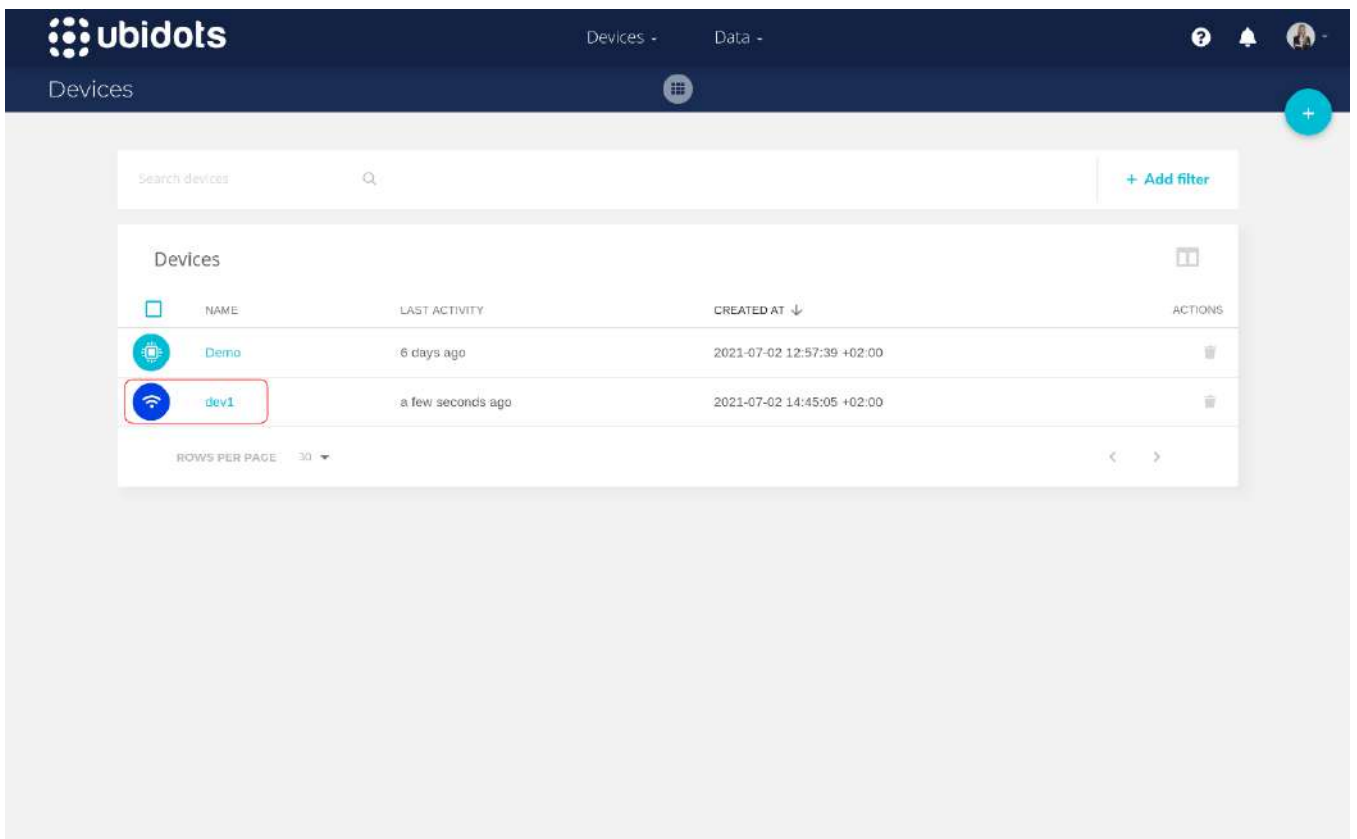
Paste the Plugin ID and Ubidots token values from Ubidots.



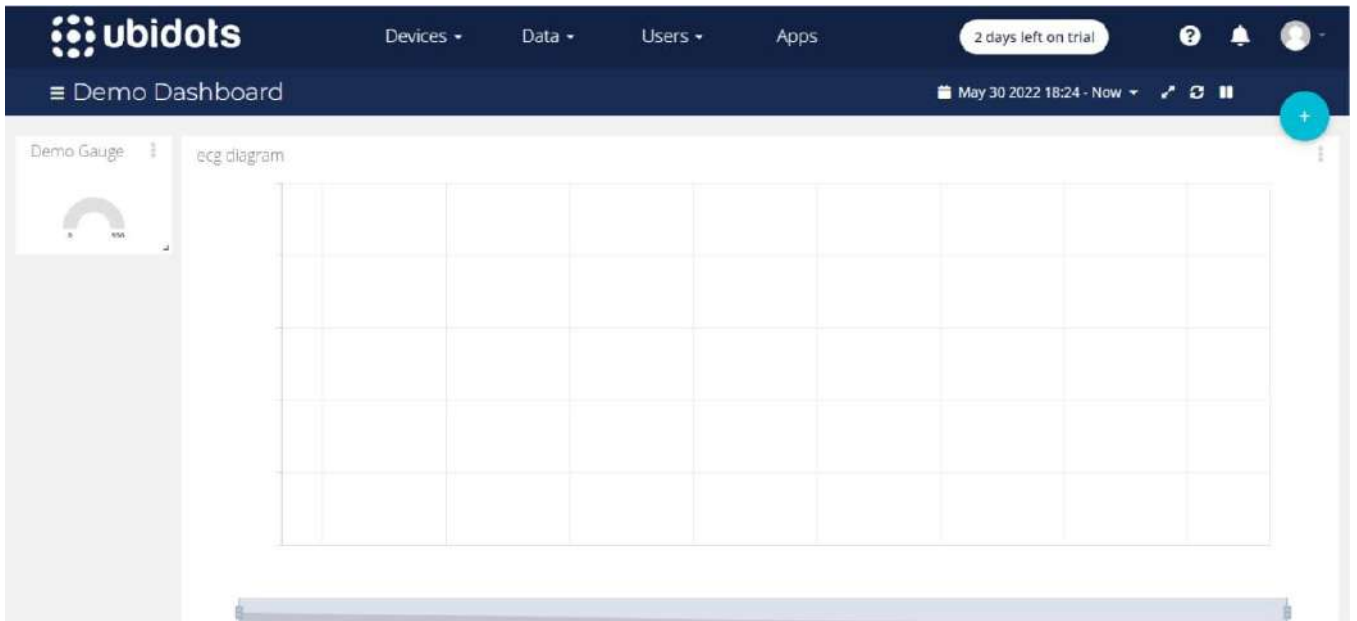
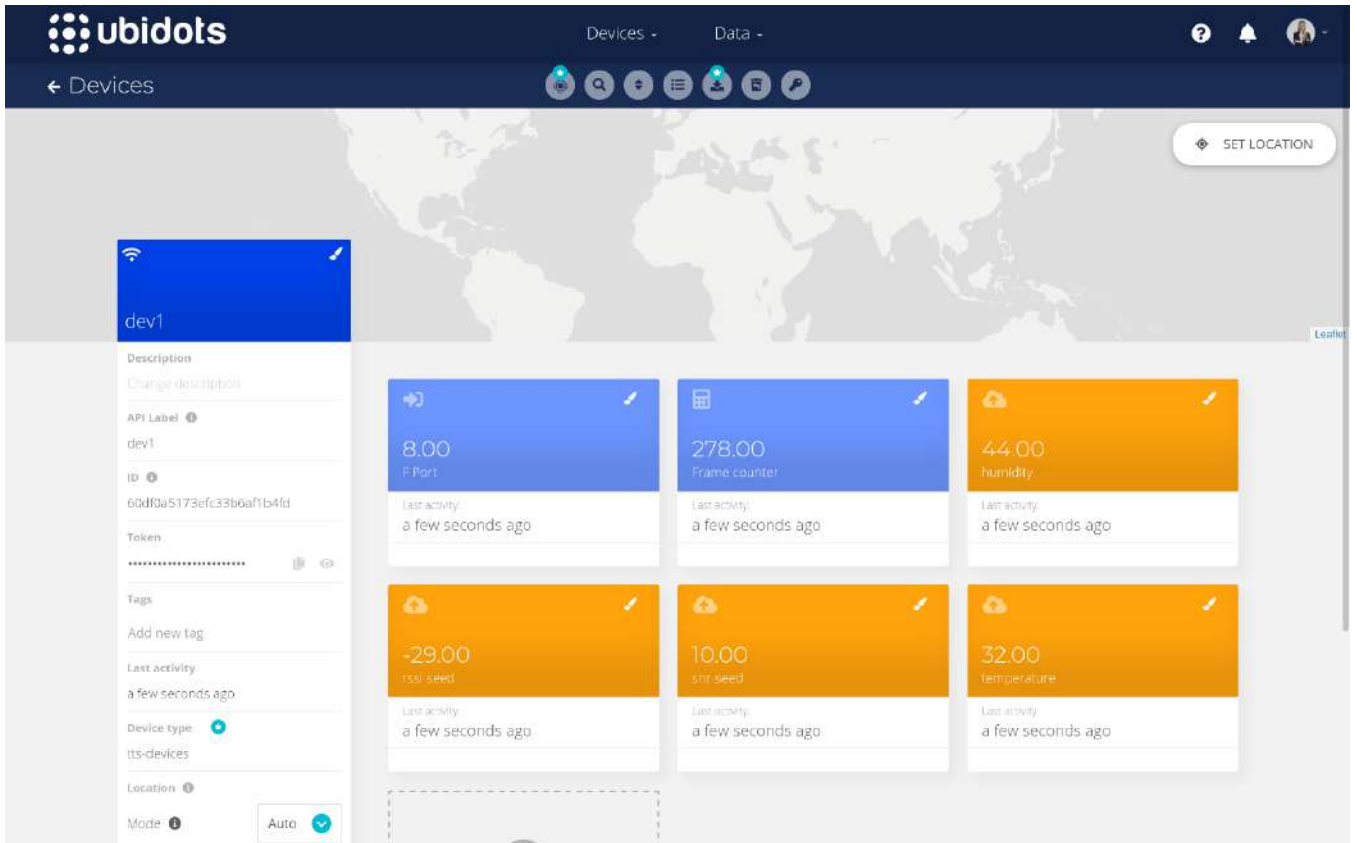
To see the values of all parameters of the Ubidots integration, click on the integration after you created it with the Webhook template.

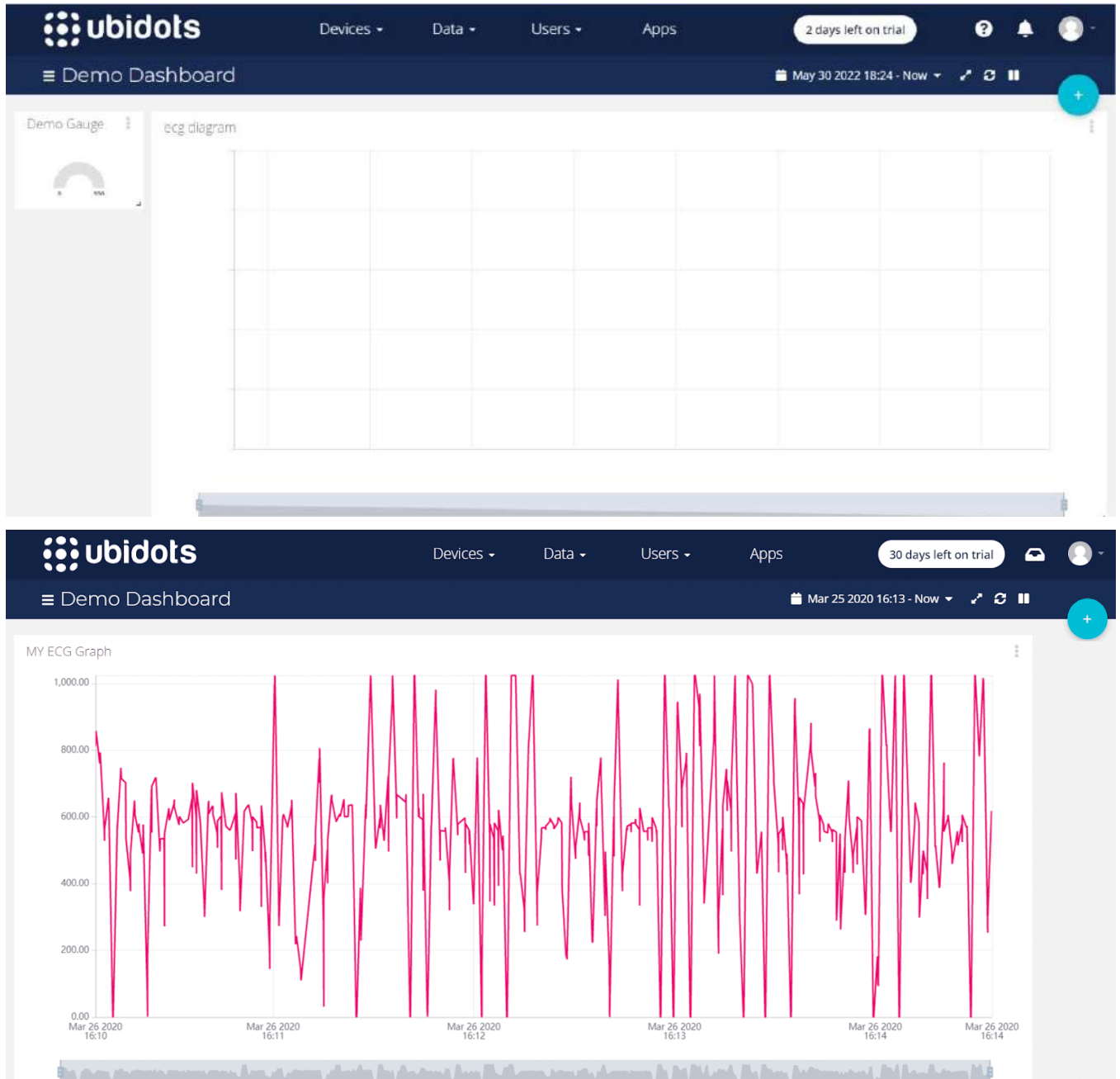
3.2.4.1.2 Monitor Your Data

After you have completed the integration, navigate to Devices menu. You will see your end device appearing in the end devices list as soon as it sends an uplink message.



Select your device to enter its dashboard. Some variables are automatically loaded from uplink metadata, while some are loaded as a payload decoder result (like humidity and temperature).





3.2.4.2 IoT based ECG Monitoring with AD8232 ECG Sensor & ESP8266

you can visualize the ECG waveform on the Serial Plotter Screen. But now we want to visualize the ECG waveform remotely from any part of the world. So for that, I won't need to send the signal generated to any IoT platform. For that I used Ubidots. Using Ubidots you can send data to the cloud from any Internet-enabled device [33].

Chapter four: Proposed Method & Results & Dissociation

1. Introduction

In this chapter, we discussed the method of connecting the device and the method of sending an alert to the doctor, as well as discussing different cases.

2. Proposed Method

1. The procedure

1. we have used three sensors that made the home monitoring patient system possible to operate, first of all, is the MLX90614 sensor which measures the temperature of the human body using infrared rays without the need to be in contact with the object, the second one is the DHT21

sensor which measures the atmosphere (temperature and the humidity) of the patient room and finally, the third sensor which is kAD8232 ECG Sensor, this sensor measures the heartbeat rate and displays the ECG diagram so that the doctor can keep an eye on the patient.

2. We connected these three sensor with the node MCU esp8266 which is a microcontroller that allows communications via the internet besides all the features that any microcontroller provides like controlling the functions of embedded systems.
3. the final results that we get are the patient's temperature, heart rate, and ECG diagram, and the room temperature and humidity so that the doctor who's been assigned to the patient can look after them.

2. ESP8266 NodeMCU Send Emails using an SMTP Server

4.2.2.1 Introducing SMTP Servers

SMTP means Simple Mail Transfer Protocol and it is an internet standard for email transmission. To send emails using an ESP8266, you need to connect it to an SMTP Server [34].

4.2.2.1.1 ESP-Mail-Client Library

To send emails with the ESP8266 NodeMCU board, we'll use the ESP-Mail-Client library. This library allows the ESP8266 to send and receive emails with or without attachments via SMTP and IMAP servers [34].

4.2.2.1.1.1 Installing the ESP-Mail-Client Library

You need to install the ESP-Mail-Client library. This library is not available to install through the Arduino IDE Library Manager. Follow the next steps to install the library [34]:

1. Click here to download ESP-Mail-Client library .zip folder.
2. In your Arduino IDE, go to Sketch > Include Library > Add .ZIP Library.
3. Select the .zip file you've just downloaded.

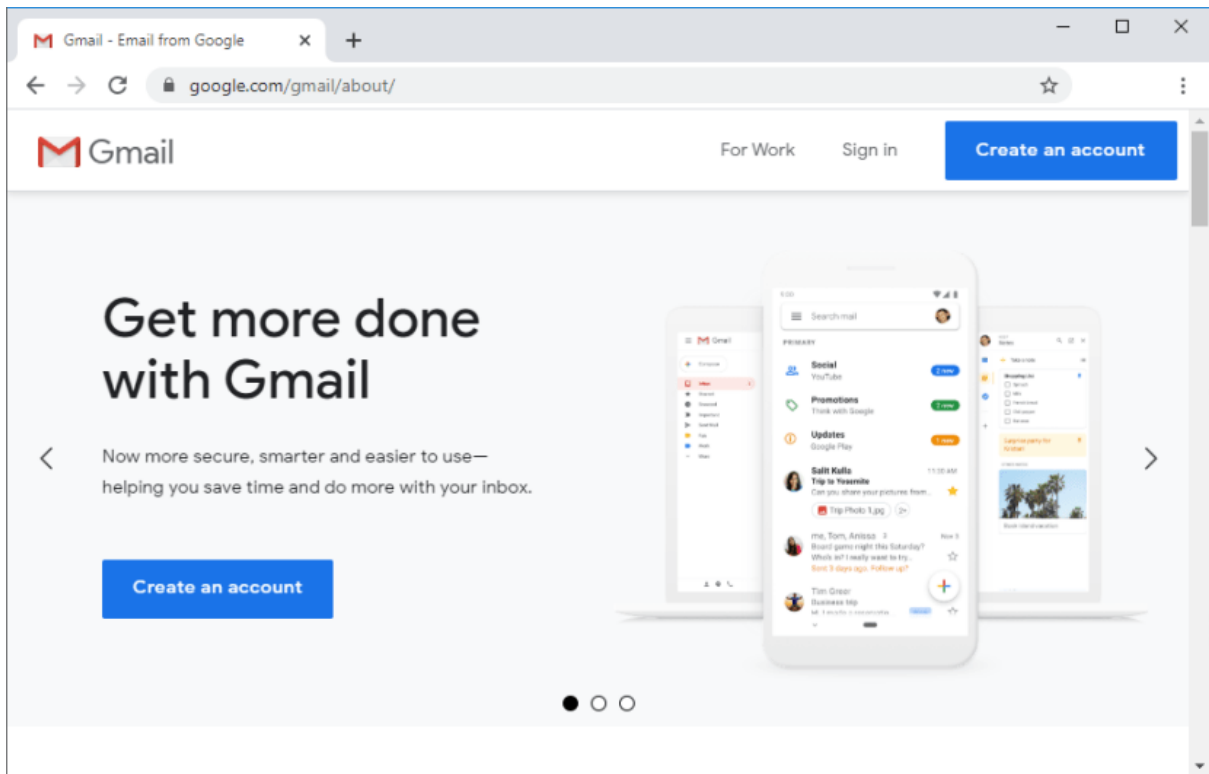
4.2.2.2 Sender Email (New Account)

We recommend creating a new email account to send the emails to your main personal email address. Do not use your main personal email to send emails via ESP8266. If something goes wrong in your code or if you make too many requests by mistake, you can be banned or have your account temporarily disabled.

We'll use a newly created Gmail.com account to send the emails, but you can use any other email provider. The receiver email can be your personal email without any problem [34].

4.2.2.2.1 Create a Sender Email Account

we create a new email account for sending emails with the ESP8266.



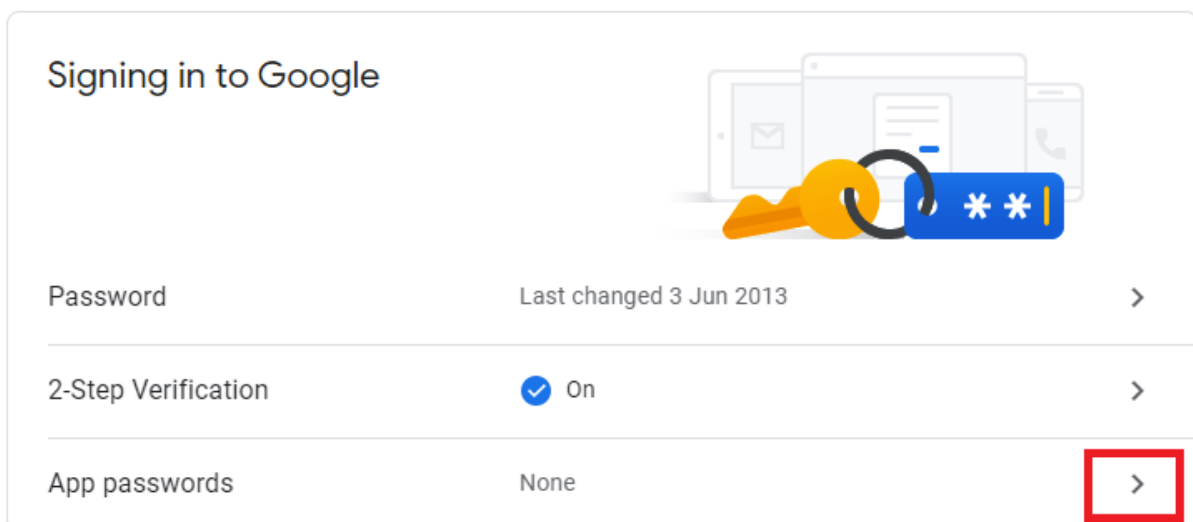
You need to create an app password so that the ESP32 is able to send emails using your Gmail account. An App Password is a 16-digit passcode that gives a less secure app or device permission to access your Google Account. Learn more about sign-in with app passwords here.

An app password can only be used with accounts that have 2-step verification turned on.

1. Open my Google Account.
2. In the navigation panel, we select Security.
3. Under “Signing in to Google,” we select 2-Step Verification > Get started.
4. Follow the on-screen steps.

After enabling 2-step verification, we can create an app password.

1. Open my Google Account.
2. In the navigation panel, we select Security.
3. Under “Signing in to Google,” select App Passwords.



4. In the Select app field, choose mail. For the device, select Other and give it a name, for example, ESP32. Then, click on Generate. It will pop up a window with a password that you'll use with the ESP32 or ESP8266 to send emails. Save that password because you'll need it later.

Generated app password

Your app password for your device

XXXXXXXXXXXXXXXXXXXX

How to use it

Go to the settings for your Google Account in the application or device you are trying to set up. Replace your password with the 16-character password shown above. Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone.

DONE

Email


Password

Now, we have an app password that you'll use on the ESP32 code to send the emails.

← App passwords

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. [Learn more](#)

Your app passwords

| Name | Created | Last used | |
|---------|---------|-----------|---|
| ESP8266 | 11:33 | - |  |

Select the app and device for which you want to generate the app password.

Select app ▼ Select device ▼

GENERATE

If we're using another email provider, check what we need to do to allow less secure apps. we should be able to find the instructions with a quick google search "my_email_provider + allow less secure apps".

2. Gmail SMTP Server Settings

If we're using a Gmail account, these are the SMTP Server details:

- SMTP Server: smtp.gmail.com
- SMTP username: Complete Gmail address
- SMTP password: Your Gmail password
- SMTP port (TLS): 587
- SMTP port (SSL): 465
- SMTP TLS/SSL required: yes

4.2.2.2.3 Outlook SMTP Server Settings

For Outlook accounts, these are the SMTP Server settings:

- SMTP Server: smtp.office365.com
- SMTP Username: Complete Outlook email address
- SMTP Password: Your Outlook password
- SMTP Port: 587
- SMTP TLS/SSL Required: Yes

4.2.2.2.4 Live or Hotmail SMTP Server Settings

For Live or Hotmail accounts, these are the SMTP Server settings:

- SMTP Server: smtp.live.com
- SMTP Username: Complete Live/Hotmail email address
- SMTP Password: Your Windows Live Hotmail password
- SMTP Port: 587
- SMTP TLS/SSL Required: Yes

In my Arduino IDE, we go to File > Examples > ESP-Mail-Client and experiment with the examples provided—we need to insert our email details (sender and recipient accounts), the SMTP server settings, and your SSID and password [34].

3. Send an Email with HTML or Raw Text with ESP8266

The following code sends an email via SMTP Server with HTML or raw text. As an example, the ESP8266 sends an email once when it boots. Then, we should be able to modify the code and integrate it into our own projects [34].

```
#include <Arduino.h>
#if defined(ESP32)
  #include <WiFi.h>
#elif defined(ESP8266)
  #include <ESP8266WiFi.h>
#endif
#include <ESP_Mail_Client.h>
```

```
SMTPSession smtp;

void smtpCallback(SMTP_Status status);
void setup(){
  WiFi.begin( "Mohmed_1" , "AMILC2021");
  while (WiFi.status() != WL_CONNECTED){ delay(10);}

  smtp.debug(1);
  smtp.callback(smtpCallback);
  ESP_Mail_Session session;
  session.server.host_name = "smtp.gmail.com";
  session.server.port = 465;
  session.login.email = "aishakareem199@gmail.com";
  session.login.password = "xvznpohqmotkwxat";
  session.login.user_domain = "";
  SMTP_Message message;
  message.sender.name = "H.M.P.S.";
  message.sender.email = "aishakareem199@gmail.com";
  message.subject = "ESP Test Email";
  message.addRecipient("Wafa Hassan", "taheralkarbalaie@gmail.com");

  String textMsg = "ALARM - Sent from ESP board";
  message.text.content = textMsg.c_str();
  message.text.charSet = "us-ascii";
  message.text.transfer_encoding = Content_Transfer_Encoding::enc_7bit;

  smtp.connect(&session);
  MailClient.sendMail(&smtp, &message);}
  void loop(){ }
  void smtpCallback(SMTP_Status status){ }
```

we need to insert our network credentials, as well as set the sender email, SMTP Server details, recipient, and message [34].

3. Device connection

4. Arduino code

| | |
|---|---|
| <pre>#include <ESP8266WiFi.h> #include <PubSubClient.h> #define timer_value 10000 #include <Wire.h> #include <LiquidCrystal_I2C.h> LiquidCrystal_I2C lcd(0x3F,20,4); #include <DHT.h> DHT dht(2, DHT21); #include <Adafruit_MLX90614.h> Adafruit_MLX90614 mlx = Adafruit_MLX90614(); #include <Arduino.h> #if defined(ESP32) #include <WiFi.h> #elif defined(ESP8266) #include <ESP8266WiFi.h> #endif #include <ESP_Mail_Client.h> SMTPSession smtp; long instance1=0, timer; double hr = 72, interval = 0; int value = 0, count = 0; bool flag = 0; char mqttBroker[] = "industrial.api.ubidots.com"; char payload[100]; char topic[150]; char str_sensor[10]; WiFiClient ubidots; PubSubClient client(ubidots); void smtpCallback(SMTP_Status status); unsigned long t1 , t2 , t3 , t4 , t5 , t6; float x , a , z , temperature , humidity;; void callback(char* topic, byte* payload, unsigned int length) { char p[length + 1]; memcpy(p, payload, length); p[length] = NULL;</pre> | <pre>else { digitalWrite(10, 1); value = analogRead(A0); value = map(value, 250, 400, 0, 100); } temperature = dht.readTemperature(); humidity = dht.readHumidity(); x = mlx.readObjectTempC() ; a = x*100; digitalWrite(10, 1); value = analogRead(A0); value = map(value, 250, 400, 0, 100); hr = count*6; int y = map(a,2700,3200,360,410); //y=constrain(y,360,410); z= y/10.00 ; if((x >=1000) && (value > 100)){ x=temperature*1.05 ;} else{ if(millis(>t2+1000){ lcd.clear(); lcd.setCursor(0,0); lcd.print("Patient temp.= "); if(x<27){lcd.print(x) ;} else{lcd.print(z) ; } lcd.setCursor(0,1); lcd.print("Room temp.= "); lcd.print(temperature); lcd.setCursor(0,2); lcd.print("Hum(%)= "); lcd.print(humidity); lcd.setCursor(0,3); lcd.print("hr= "); lcd.print(hr); t2=millis();} smtp.debug(1); smtp.callback(smtpCallback); ESP_Mail_Session session; session.server.host_name = "smtp.gmail.com"; session.server.port = 465; session.login.email =</pre> |
|---|---|

| | |
|---|---|
| <pre> Serial.write(payload, length); } void reconnect() { while (!client.connected()) { Serial.println("Attempting MQTT connection..."); if (client.connect("myecgsensor" , "BBFF- 2aPwtlXmJUU34iqqgAUyTG7InnHzPC", "")) { Serial.println("Connected"); } } } void setup() { Serial.begin(115200); WiFi.begin("tf_414", "44332211"); pinMode(A0, INPUT); pinMode(14, INPUT); pinMode(12, INPUT); while (WiFi.status() != WL_CONNECTED) { delay(50); } client.setServer(mqttBroker, 1883); client.setCallback(callback); mlx.begin(); dht.begin(); lcd.init(); lcd.backlight();} void loop() { if (!client.connected()) {reconnect(); } sprintf(topic, "%s%s", "/v1.6/devices/", "esp8266"); sprintf(payload, "%s", ""); // Cleans the payload sprintf(payload, "{\"%s\":", "myecg"); float myecg = analogRead(A0); dtostrf(myecg, 4, 2, str_sensor); sprintf(payload, "%s {\"value\": %s}", payload, str_sensor); </pre> | <pre> "aishakareem199@gmail.com"; session.login.password = "zvznpohqmotkwxat"; session.login.user_domain = ""; SMTP_Message message; message.sender.name = "H.M.P.S."; message.sender.email = "aishakareem199@gmail.com"; message.subject = "ESP Test Email"; message.addRecipient("Taher Falah", "taheralkarbalaie@gmail.com"); String x1 = "patient temperature is not normal"; String x2 = "room temperature is not normal"; String x3 = "room humidity is not normal"; String x4 = "patient heart rate is not normal"; if(z >39){ message.text.content = x1.c_str(); if(millis()>t3+3000) {MailClient.sendMail(&smtp, &message); t3=millis();} } if(temperature >25.5){ message.text.content = x2.c_str(); if(millis()>t4+3000) {MailClient.sendMail(&smtp, &message); t4=millis();} } if(humidity >35){ message.text.content = x3.c_str(); if(millis()>t5+3000) {MailClient.sendMail(&smtp, &message); t5=millis();} } if(hr >77){ message.text.content = x4.c_str(); if(millis()>t6+3000) {MailClient.sendMail(&smtp, &message); t6=millis();} } message.text.charSet = "us-ascii"; message.text.transfer_encoding = </pre> |
|---|---|

| | |
|--|---|
| <pre>Serial.println("Publishing data to Ubidots Cloud"); client.publish(topic, payload); client.loop(); if((digitalRead(10) == 1) ((digitalRead(11) == 1)){ digitalWrite(10, 0); instance1 = micros(); timer = millis();}</pre> | <pre>Content_Transfer_Encoding::enc_7bit; smtp.connect(&session); }} void smtpCallback(SMTP_Status status){ }</pre> |
|--|---|

3. Dissociation

4.3.1 Body temperature

The normal body temperature of a person varies depending on gender, recent activity, food and fluid consumption, time of day, and, in women, the stage of the menstrual cycle. Normal body temperature can range from 97.8 degrees F (or Fahrenheit, equivalent to 36.5 degrees C, or Celsius) to 99 degrees F (37.2 degrees C) for a healthy adult. A person's body temperature can be taken in any of the following ways:

- Orally. Temperature can be taken by mouth using either the classic glass thermometer, or the more modern digital thermometers that use an electronic probe to measure body temperature.
- Rectally. Temperatures taken rectally (using a glass or digital thermometer) tend to be 0.5 to 0.7 degrees F higher than when taken by mouth.
- Axillary. Temperatures can be taken under the arm using a glass or digital thermometer. Temperatures taken by this route tend to be 0.3 to 0.4 degrees F lower than those temperatures taken by mouth.
- By ear. A special thermometer can quickly measure the temperature of the ear drum, which reflects the body's core temperature (the temperature of the internal organs).
- By skin. A special thermometer can quickly measure the temperature of the skin on the forehead.

Body temperature may be abnormal due to fever (high temperature) or hypothermia (low temperature). A fever is indicated when body temperature rises about one degree or more over the normal temperature of 98.6 degrees Fahrenheit, according to the American Academy of Family Physicians. Hypothermia is defined as a drop in body temperature below 95 degrees Fahrenheit [35].

4.3.1.1 How the MLX90614 (Infrared thermometer) works

Most of the temperature measurement techniques around the world require some sort of physical contact between the temperature sensor and the object or environment whose temperature is to be measured, but as technology advanced, this changed too. The need to be able to measure the temperature of an object without physical contact arose. This need brought the measurement of temperature using infrared sensors. The principle of operation of Infrared thermometers is simple, all bodies at a temperature above 0 Kelvin (absolute zero) emit an infrared energy which can be detected by the infrared 29 thermometer sensor. Its design includes a lens that focuses the infrared energy being emitted by the object in front of a detector. The detector converts the energy into an electrical signal which then can be passed to a microcontroller to interpret and display in units of temperature after compensating for the variation in ambient temperature[36].

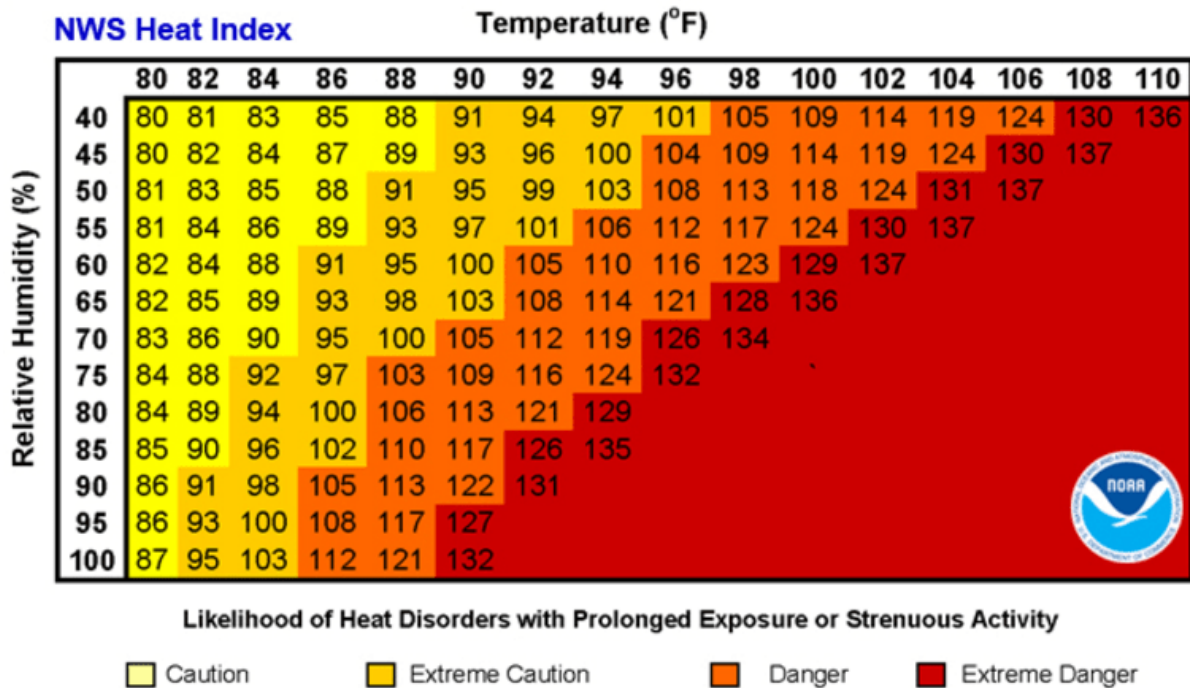
4.3.2 Humidity

1. Why are we so sensitive to humidity?

Humidity is how much water vapor is in the air. When we perspire, our bodies normally rely on air to get rid of the sweat that accumulates on the skin. This allows the body to cool down. When the humidity in the air is high, the warm moisture stays on our skin longer, making us feel even hotter. Meteorologists call this the “heat index”.

According to the National Weather Service, the heat index is a measure of how hot it really feels when humidity is factored in with the actual temperature.

the NWS Health Index chart, you can see that temp/humidity combo is putting your body in danger of suffering from a heat disorder [37].



National Weather Service heat index chart

2. Health Risks of High Humidity

High humidity can have a number of adverse effects on the human body. It can contribute to feelings of low energy and lethargy. In addition, high humidity can cause hyperthermia — overheating as a result of your body’s inability to effectively let out heat. Hyperthermia can cause [37]:

- Dehydration
- Fatigue
- Muscle cramps
- Heat exhaustion
- Fainting
- Heat stroke

4.3.3 Heart Rate

A strong healthy heart will speed up and slow down constantly to address environmental changes and stressors as they appear. Consistency is not something to strive for when it comes to the heart. Healthy hearts should have beat-to-beat variability and not consistently beat at the same tempo. This variation between heartbeats is called Heart Rate Variability (HRV).

Looking at HRV can provide a lot of information about an individual's physical and mental health because it demonstrates the body's ability to respond to stress and change. Higher HRV generally means the body is better prepared and resilient when faced with adversities.

HRV is calculated by looking at the fluctuations between individual heartbeats. The time between each consecutive heartbeat has many names such as R-R interval or inter-beat-interval (IBI) and is calculated in milliseconds. That data is then used to calculate the variability that exists in the person's heartbeat. Using HRV, an individual's Autonomic Nervous System (ANS) can be analyzed [38].

What can cause heart rate to change?

As well as exercise, there are many other reasons why your heart rate may suddenly change. Some common reasons include illness or fever, dehydration, anxiety, medications, as well as other health conditions [39].

Dehydration

When you are dehydrated, the volume of your blood decreases and your heart needs to work harder to pump blood around your body. You may notice a faster heart rate and palpitations (a feeling of being aware of your heartbeat, or that your heart is pounding or beating irregularly).

Staying hydrated is an important way to regulate your heart rate and has many other health benefits. Women should aim to drink 6-8 glasses of water per day (or 1.6 litres), and men 8-10 glasses (or 2 litres).

Infection or fever (including Covid-19)

It's common for your pulse rate to increase in response to an infection or fever. This happens as your heart pumps harder to deliver oxygen and immune cells around your body, helping to fight the infection. Viral infections such as the flu or Covid-19 may cause your heart rate to increase.

Medication

Some medications can also cause changes to your heart rate. For example, certain asthma medications can cause your heart rate to speed up, while heart medications (like beta blockers) can make your heart rate slow down.

If you notice sudden changes to your pulse rate after starting a new medication, and you are not feeling well, it's best to speak to your GP.

Heart conditions

In some cases, an abnormal heart rhythm (arrhythmia) can be caused by a problem with your heart's electrical conduction system. This can cause your heart to beat more slowly, quickly, or in an irregular way. Some arrhythmias are more serious than others, and some may be related to more serious heart conditions [39].

- Keep in mind that many factors can influence heart rate, including [40]:
- Age
- Fitness and activity levels
- Being a smoker
- Having cardiovascular disease, high cholesterol or diabetes
- Air temperature
- Body position (standing up or lying down, for example)
- Emotions
- Body size

➤ Medications

4.3.3.1 Child's heart rate healthy

Your child's heart rate (also called pulse) can vary wildly throughout the day. Heart rate is the number of times the heart beats each minute. Daily activities can change how fast or slow the rate fluctuates – from a slow, steady beat while resting or sleeping to a higher rate during exercise.

"There's a wide variation in what a normal heart rate can be depending on the age of the child as well as the biological make-up of that individual child," says Colin Kane, M.D., pediatric cardiologist at Children's HealthSM and Director of the Cardiology Outreach Program. "Even kids who are the same age can have different resting heart rates."

➤ healthy heart rate for a child

When the child is sitting quietly, their heart rate is considered a resting heart rate. A healthy resting heart rate can vary by age.

Newborns 0 to 1 month old: 70 to 190 beats per minute

Infants 1 to 11 months old: 80 to 160 beats per minute

Children 1 to 2 years old: 80 to 130 beats per minute

Children 3 to 4 years old: 80 to 120 beats per minute

Children 5 to 6 years old: 75 to 115 beats per minute

Children 7 to 9 years old: 70 to 110 beats per minute

Children 10 years and older: 60 to 100 beats per minute

It's likely that child's pulse stays within these healthy ranges, even if the pulse feels very fast. Understanding the variations in heart rates and how to properly check a child's rate can help keep track and prevent unnecessary concern [41].

"Your child's heart rate is typically not linked to an intrinsic heart problem," says Dr. Kane. "Their heart rate can go up with anything that makes them excited or uncomfortable. When this happens, it's just a natural response to stress."

Table 1 Pediatric heart rate lower limit, normal range, and upper limit by age [42].

| Age | Lower limit (1 st percentile) | Normal range (10 th to 90 th percentile) | Upper limit (99 th percentile) |
|------------------|---|---|--|
| 0 to 3 months | 107 | 123 to 164 | 181 |
| 3 to <6 months | 104 | 120 to 159 | 175 |
| 6 to <9 months | 98 | 114 to 152 | 168 |
| 9 to <12 months | 93 | 109 to 145 | 161 |
| 12 to <18 months | 88 | 103 to 140 | 156 |
| 18 to <24 months | 82 | 98 to 135 | 149 |
| 2 to <3 years | 76 | 92 to 128 | 142 |
| 3 to <4 years | 70 | 86 to 123 | 136 |
| 4 to <6 years | 65 | 81 to 117 | 131 |
| 6 to <8 years | 59 | 74 to 111 | 123 |
| 8 to <12 years | 52 | 67 to 103 | 115 |
| 12 to <15 years | 47 | 62 to 96 | 108 |
| 15 to 18 years | 43 | 58 to 92 | 104 |

- A child might have a fast heart rate if they are:
- Playing or exercising vigorously

- Experiencing pain
- Feeling anxious or stressed
- Experiencing a fever or illness
- Drinking a lot of caffeine or energy drinks
- Dehydrated

If your child is experiencing any of the above, a fast heart rate is typically not a cause for concern, though drinking a lot of caffeine can cause problems in some children. Also, remember that your child's heart naturally beats faster than an adult heart and can get much faster during exercise than an adult heart rate [41].

However, if your child is experiencing symptoms such as chest pain or trouble breathing along with a fast heart rate, they may need medical attention. Dr. Kane says a good rule of thumb is if your child's heart is beating too fast for you to count the beats, then medical help may be needed.

A child typically experiences a slower heart rate when sleeping. However, if their heart rate is slow in the middle of the day and they show symptoms of lethargy or experience fainting.

4.3.3.2 Biological Sex in Normal Cardiac Function and in its Disease

- Biological sex plays an important role in normal cardiac physiology as well as in the heart's response to cardiac disease. Women generally have better cardiac function and survival than do men in the face of cardiac disease; however, this is progressively lost when comparing postmenopausal women with age matched men. Animal model of cardiac disease mirror what is seen in humans. Sex hormones contribute significantly to sex based difference in cardiac functioning and in its disease outcome. Estrogen is considered to be cardioprotective, whereas testosterone is detrimental to heart function.
- Large studies have been carried out with respect to the cardiac functions among the different sexes in both animals and humans, though the studies on male predominate. There is increasing evidence that the biological sex; play a significant role in cardiac functioning, as well as in the occurrence and outcome of cardiac disease. Female sex tends to have an upper hand in the heart function which is progressively lost once they attain menopause. Sex differences have been reported in left ventricular hypertrophy, cardiac remodelling with aging, arrhythmogenic activity and post-infarct myocardial salvage. This has led to the hypothesis that the female sex hormone estrogen has a cardio protective mechanism which in turn is responsible for sex specific differences in the normal cardiac function and in its disease outcome.

4.3.3.2.1 Normal structure and function of human heart

There is no significant difference in the cardiac size between the male and female until the onset of puberty. This proves that the number of cardiac myocytes is same in both the sexes [43]. Further, it is widely accepted that postnatal growth primarily occurs by an increase in cardiac myocyte size as the tissue lose their ability to proliferate shortly after birth [44]. It is seen that after puberty there is a 15-30% increase in the heart mass in males which is proportionate to the body size. Thus the male myocyte undergoes a greater degree of hypertrophy when compared to female. The myocyte hypertrophy is proportionally symmetrical throughout the heart suggesting that there is no sex difference with respect to relative wall thickness of the heart [43].

The human heart beats approximately 70 to 85 times per minute in an average adult, with a notable difference between the genders. The average adult male heart rate is between 70 and 72 beats per minute, while the average for adult women is between 78 and 82 beats. This difference is largely accounted for by the size of the heart, which is typically smaller in females than males. The smaller female heart, pumping less blood with each beat, needs to beat at a faster rate to match the larger male heart's output. Further women have a different intrinsic rhythmicity to the pacemaker of their hearts, which causes them to beat faster [45].

As age advances there is an increase in both septal and wall thickness in both sexes, however the left ventricular diameter Physiology Section increases only in males [46]. The above result suggests that there is a loss of myocardial mass in males but not in females. As men ages they lose an average of 1gm of cardiac mass everyday, which is approximately equivalent to 64 million cardiac myocytes. This loss is attributed to compensatory hypertrophy in men to maintain adequate heart mass. Unlike in women the myocyte number and size is preserved with ageing [47]. With respect to the cardiac function, women have a better diastolic function than men, however this reduces with ageing in both the sexes [46]. In addition, men have decreased systolic function than women as age advances. Varying results have been observed with respect to exercise performance and cardiac adaptation in men and women. Studies reveal that there is an increase in cardiac output during exercise in both the sexes [48]; however different mechanisms coexist [49]. Well-trained athletes are known to have slower heart rates than non-athletes. The heart, as a muscle, improves its strength as a result of exercise training, particularly with aerobic training. The heart rate in female athletes and regular exercisers will be lower than that of an untrained male; however it will still beat at a faster rate than an equally trained male athlete or regular exerciser [45].

2. Diseases

Women tend to experience heart disease 10yr later than men. Cardiovascular disease outcome has revealed the fact that premenopausal women fare better than men, they tend to have better prognosis than men in response to diseases like hypertension; aortic stenosis and hypertrophic cardiomyopathy (HCM). Women with congestive heart failure tend to survive better than men [50]. Also, following an ischemic event, women tend to have a better long term prognosis, even though their immediate death rate is elevated, when compared to men [51]. Another study, however reports no difference in congestive heart failure prognosis for men and women [52]. Thus, cardiac function is preserved in women, whereas men experience poor cardiac contractility owing to chamber dilatation and wall thinning [53].

Also, studies from National Centre for health statistics reports that age also plays a major role in cardiovascular diseases. Upto to 64yr, men with cardiovascular diseases die earlier than women but after 65yr the death rate is higher in women than in men. Further the above studies reports a greater prevalence of congestive heart failure in men than women aged between 65-74 yrs but no difference after the age of 74 [54]. The discrepancies in the outcome of the above studies have attributed to the fact that women have been underrepresented in clinical trial [55] which may be due to their poor enrolment. Further women generally do not develop cardiovascular disease until older age.

However, women do not always fare better than men. They tend to have a poor prognosis in case of dilated cardiomyopathy [56]. Women are more sensitive to alcohol induced cardiac disease [57] than men, which may or may not be attributed to estrogen. Further, women are more likely to die after their first heart attack which may be because the symptoms of heart attack are different in women when compared to men. This is due to doctors and patients often attribute chest pains in women to noncardiac causes, leading to misinterpretation of their condition. Men usually experience crushing chest pain during a heart attack. Women may have a greater tendency to have pain just under the breastbone, or complain of abdominal pain, indigestion, difficulty breathing, nausea and unexplained fatigue.

4.3.3.3 ECG lead placement

To ensure an accurate ECG reading, it's important to correctly place the sensor leads. The three-lead EKG cable has three leads colors in red, green, and yellow.

- The red electrode must be placed under the right clavicle of the subject, which is near the right shoulder and within the rib cage frame.
- The yellow electrode must be placed under the left clavicle of the subject, which is near the left shoulder and within the rib cage frame.

- The green electrode must be placed on the left side of the subject, below the pectoral muscles, near the lower edge of the left rib cage.

For the best results, all of the leads must be placed on the chest wall equidistant from the heart.

Alternatively, the pads can be placed on the limbs — the red electrode on the right arm, the yellow electrode on the left arm, and the green electrode on the right leg [58].

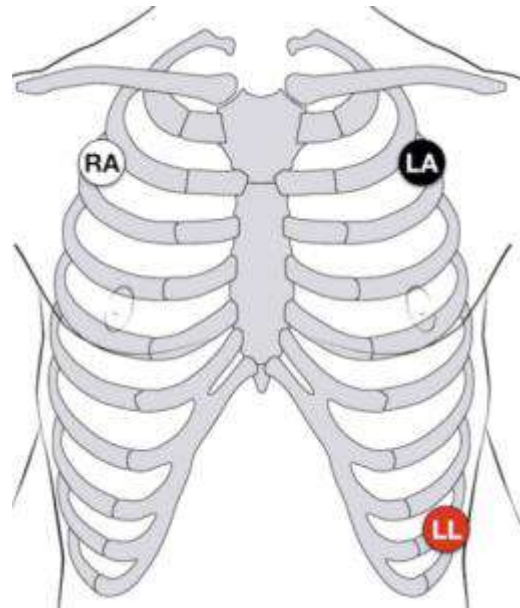


Figure The three-electrode ECG system is illustrated here.

4.3.3.4 How it works

ECG is the graphical recording of the electrical signals of the heart. It is useful in determining a person's heart rate and rhythm. It serves as an indicator of cardiac electrical activities and may point to certain heart-related issues.

The ECG graph contains repeating cycles. Each cycle is divided into three parts: P, QRS, and T waves.

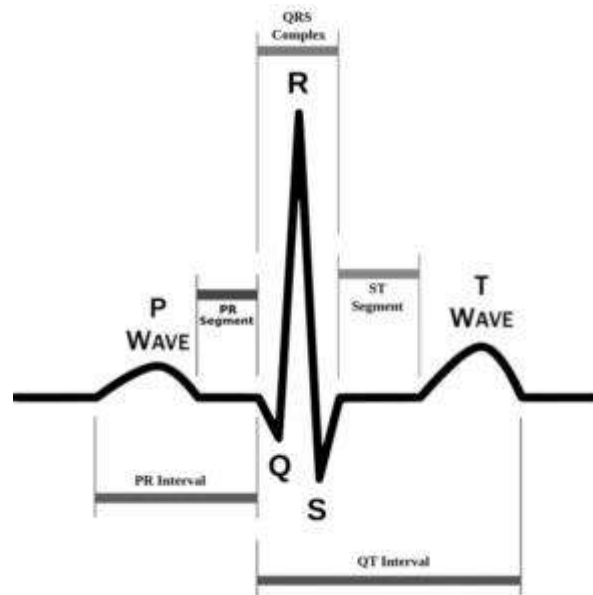
A P wave offers information about the impulse propagation time to the heart's two atria. This is followed by a flat trend, called a PR segment, which indicates the electrical impulse propagation from the atria to the ventricles.

What follows is a complex QRS wave, comprising of a small Q wave, a high R wave, and a small S wave. The QRS segment offers information about the ventricular systole in relation to the impulse propagation of the ventricles (Q wave).

The transmission to the whole tissue is caused by the R and S waves.

ECG also provides information about the heart's fibrillation and arrhythmias, which can be helpful for analyzing heart attacks. The ST interval is followed by the T wave, which indicates ischemia occurrences. The T segment is helpful in analyzing cardiac hypertrophy, heart attacks, and ischemia.

The ECG signal ends with a small peak, called a U wave. Different segments of the graph are useful in assessing different potential pathologies.



The AD8232 ECG sensor is useful for measuring heart rate and can be used as a portable cardiac monitor. It's optimized for use as a two or three-lead ECG/EKG machine.

This device offers analog output, which is received by Arduino's A1 pin in the circuit. The analog signal can be monitored on Arduino IDE's Serial Monitor or plotted on its Serial Plotter.

If the electrodes are correctly placed on a subject, the plotted graph provides fairly accurate results. The LO+ and LO- offer the leads-off detection [58].

5. Results

For testing ECG

1. The room should be quiet and patient should be relax.
2. The patient required to lying comfortably on a padded surface and far from metal materials.
3. His/her legs placed flatly, not touching one another.
4. The head of the pad prefer to be raised no more than 45 degrees.
5. The patient skin and electrodes should be clean and dry.

4. Results

In our work, we have used many sensors. First the MLX90614 (Infrared thermometer) sensor is a high-precision sensor that reads body temperature without the need for direct contact with the sensor. Another sensor that was used in this project was AD8232 ECG sensor, is a technique for gathering electrical signals which are generated from the human heart. Another sensor we used was DHT21 (humidity and temperature sensor) a high-performance temperature and humidity sensor, that provides accurate measurement, low power consumption, long-distance data transmission, automatic calibration, and long life. All sensors were connected to the ESP8266 board. All of these sensors provided a power difference based on the input parameter and these power variables were converted to output. These results are displayed on the LCD screen. An alert is sent to the doctor if the normal range for any vital signs of the patient is exceeded. In figure the measured results of the different conditions of the human body.

We have 5 reading states that the infrared sensor gives us:

The first case: If there is no object in front of the sensor, it will give us a temperature close to room temperature.

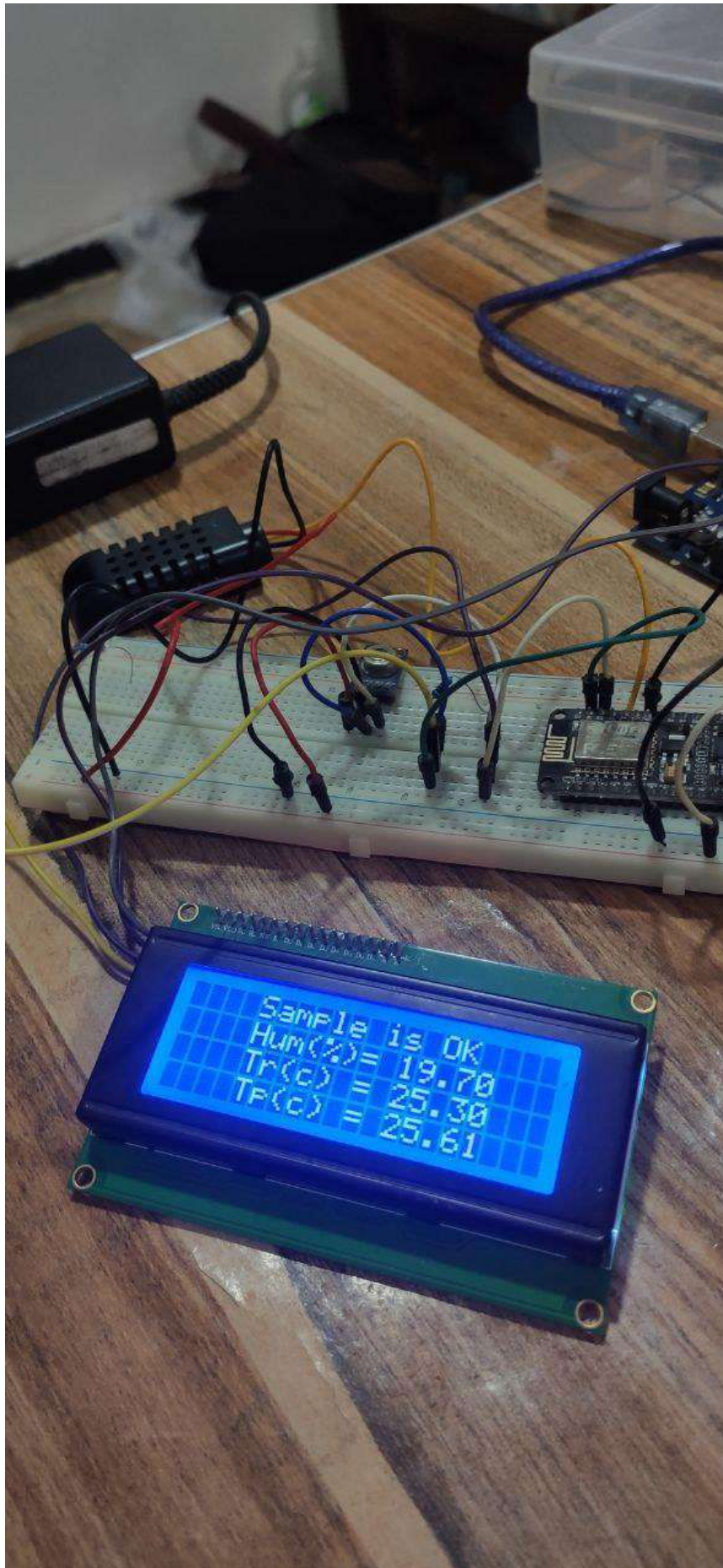


Figure no object in front of the sensor.

The second case: If there is an object in front of the sensor, but outside the reading range of the sensor, it gives us less temperature than the actual temperature of the body.



Figure object in front of the sensor and outside the reading range of the sensor.

The third case: In the event that the object is within the reading range of the sensor, which is within the limit of 2 cm to 5 cm, it gives us the correct reading of the patient's body temperature.

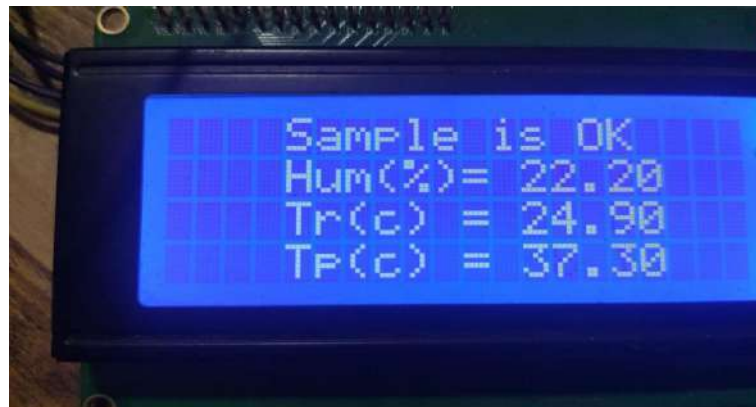


Figure object is within the reading range of the sensor.

Fourth case: If the object is present, but it is closer than the measurement range, then it gives a reading greater than the actual reading.



Figure the object is present, but it is closer than the measurement range.

The fifth case: If the object touches the sensor, here the sensor begins to give incorrect readings because it is designed to sense body temperature from a distance.

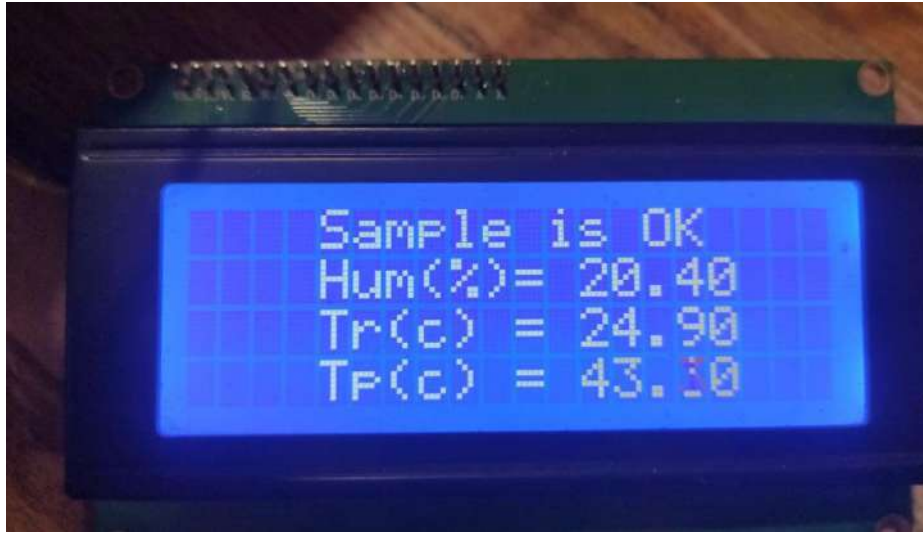


Figure the object touches the sensor.

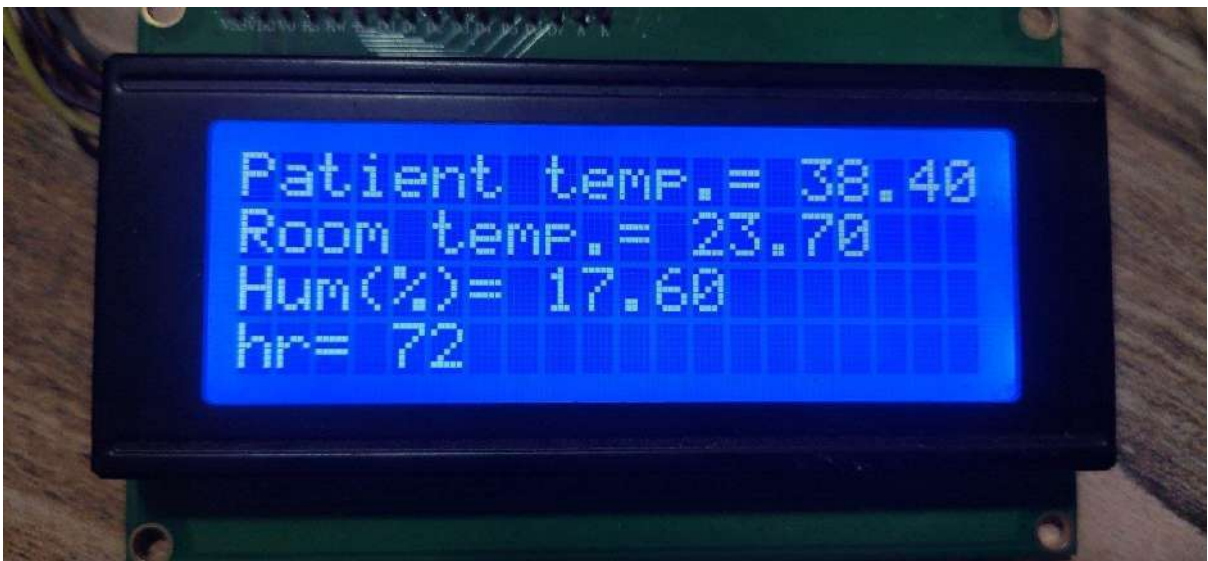


Figure measured results of different people in different environments.

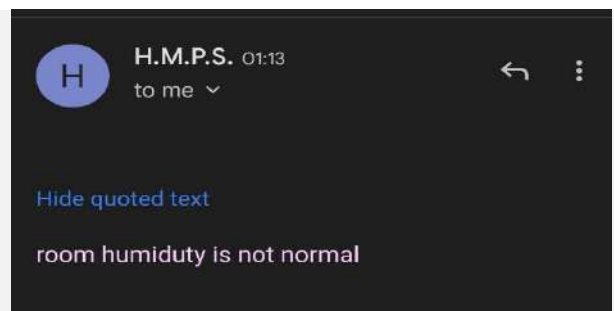
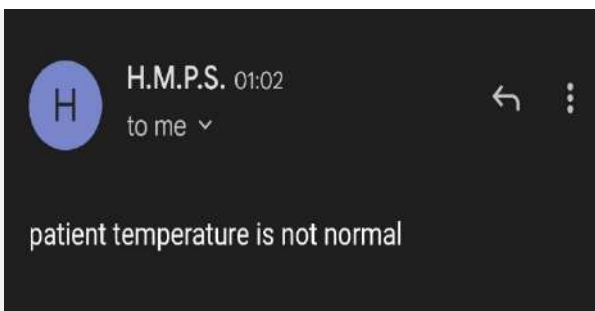
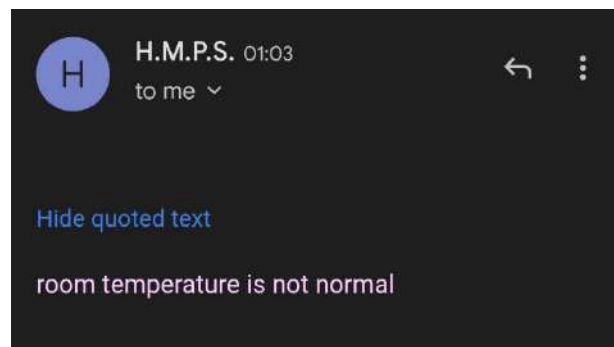
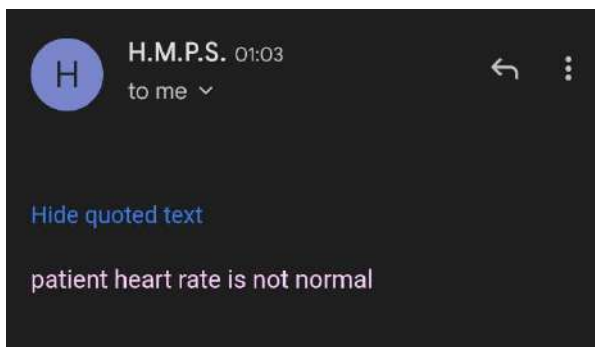


Figure send an alert to the doctor when one of the vital signs is changed.

Chapter five: Conclusion & project development for the future

5.1 Project development for the future

Telehealth is a promising technology in its effectiveness, accuracy, and spread, given the importance of medical care and monitoring in areas that lack sufficient medical staff or are difficult to access on the one hand, and as a result of the high costs of medical treatment, especially for the elderly, on the other hand. Wireless body sensor networks have received great attention from research and development, especially as a result of the tremendous development in wireless sensor networks and modern electronics that have created small-sized smart sensors that can be positioned inside or on the body there are so many ideas for developing the monitoring system if it by using new electronic chips and sensors or by updating the processor to increase the time response of the processing. When the processor chip or the microcontroller gets upgraded to a new or more powerful one like (Arduino MEGA) or Raspberry pi for high-speed response and less lag with the ability to add more sensors, and another electronic instrument. Or by using a small microcontroller like (Arduino nano) for getting smaller results as a device volume that can handle anywhere any time without having problems in the size of the device. As a big step for developing, and building an integrated system for monitoring the human condition. In addition to using an algorithm to analyze the abnormal ECG and send the result to the doctor to speed up the diagnosis and rapid intervention or use deep learning for the purpose of analysis. A blood pressure measurement system is an integrated system that works to measure the pressure of the human blood (the systolic, and the diastolic) to check the changes that happened in it. A blood pressure measurement system can be inserted into the project as a part of a device or as an extension system that can be attached to the main device. It consists of (an air pump, air valve, air pressure sensor (simple or advanced sensor), pressure cuff, processer, and small air pipes). the system work by applying air pressure by the air pump through the pipes to the cuff while the sensor measures the pressure that applies on the human arm by the cuff, and in this moment the sensor will read the systolic pressure of the blood, then the air valve will start exhaust the air out of the cuff, and through the sensor so, the sensor will take the diastolic pressure, all these steps work under the control of the microprocessor. Finally, the microprocessor will be processing the results and appearing it on to the LCD screen as a blood pressure specifying the systolic and diastolic in mmHg as a measurement unit.

5.2 Conclusion

World development and new technology appear everyday work to find new and easy solutions for medical problems that associate with disease or the early and easy diagnosis for health checks. The different types of humans wants to check the basic health parameters by using a small and mobile devices that can use in any time and any place with easy use and accurate results. This project is a real-time remote monitoring system based on the Internet of Things technology. This implementation provides an improved step-in remote health monitoring field. The numbers of people, who require health care increase year by year and the conventional monitoring systems require patients' attendance in person inside hospitals. This might cause an inefficient situation to take care of the patients, especially those who have critical and unstable health conditions. Therefore, internet technology along with modern electronic devices could offer promising solutions in this field. Based on that, this project utilizes a mobile application as an IoT platform to monitor remotely the heart rate, room temperature, room humidity, and body temperature, As well as the heart rate and send notifications to a specific smart mobile phone to be watched by a doctor. finally showing the result of this signs vital on the LCD screen and send a picture to the doctor in case the heart rate goes out of the normal range.

References

1. <https://www.glenmedsolutions.com/2017/09/21/history-patient-monitoring-systems/>
2. U. Anliker, J. A. Ward, P. Lukowicz, G.Tröster, F. Dolveck, M. Baer, F.Keita, E.Schenker, F. Catarasi, and R. Schmid, "AMON: A Wearable multi parameter medical monitoring and alerts system", IEEE Trans. On Inf. Technol., Biomed., vol. 8, no. 4, pp. 415-427, 2004.
3. Hairong Yan, HongweiHuo, "WirelessSensor Network Based E-Health SystemImplementation and Experimental Results", Member, IEEE Trans. On Consumer Electronics, vol. 56, no. 4, November 2010.
4. https://how2electronics.com/iot-ecg-monitoring-ad8232-ecg-sensor-esp8266/#IoT_based_ECG_Monitoring_with_AD8232_ECG_Sensor_038_ESP8266
5. <https://nevonprojects.com/iot-patient-health-monitoring-project/>
6. Murugesan, Malathi & Arumugam, Gowsalya & Dhanushyaa, M & Arunachalam, Janani. (2017). HOME AUTOMATION ON ESP8266. SSRG International Journal of Computer Science and Engineering. 2348-8387.
7. <https://www.dfrobot.com/blog-1337.html>
8. <https://circuitdigest.com/microcontroller-projects/iot-based-patient-monitoring-system-using-esp8266-and-arduino>
9. <https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/>
10. <https://how2electronics.com/iot-ir-thermometer-using-mlx90614-esp8266-on-blynk/>
11. <https://how2electronics.com/iot-ecg-monitoring-ad8232-ecg-sensor-esp8266/>
12. Noman Q. Al-Naggar, Husam Mohammed Al-Hammadi, Adel Mohammed Al-Fusail, and Zakarya Ali AL-Shaebi Department of Biomedical Engineering at Faculty of Engineering, University of Science and Technology, Sana'a, Yemen Correspondence should be addressed to Noman Q. Al-Naggar; noman_qaed@yahoo.com Received 11 March 2019; Revised 21 October 2019; Accepted 1 November 2019; Published 19 November 2019 Guest Editor: Bernardo Innocenti.
13. <https://www.instructables.com/Remote-Wi-Fi-DHT11-Temperature-an-Humidity-I2C-2-X/>
14. <https://www.projects8051.com/arduino-and-iot-based-patient-health-monitoring-system-project/>
15. Smith T, Sweeney R (September 2010). AnalystReport (Report). Connecticut: NERAC Publication
16. O'Donoghue J, Herbert J (2012). "Data Management within mHealth Environments: Patient Sensors, Mobile Devices, and Databases".
17. <https://blog.prevounce.com/benefits-of-remote-patient-monitoring-why-everyone-wins-with-rpm>
18. "ESP8266 Overview". Espressif Systems. Retrieved 2017-10-02.
19. ^ Brian Benchoff (August 26, 2014). "New Chip Alert: The ESP8266 WiFi Module (It's \$5)". Hackaday. Retrieved 2015-06-24.
20. ^ Brian Benchoff (September 6, 2014). "The Current State of ESP8266 Development". Hackaday. Retrieved 2015-06-24.
21. ^ "Espressif Announces ESP8285 Wi-Fi Chip for Wearable Devices". Espressif Systems. Mar 9, 2016. Retrieved 2016-07-10.
22. ^ "ESP8266 Technical Reference, Version 1.7" (PDF). Espressif Systems. Retrieved 2021-04-22.

23. ^ "Espressif ESP8266 Developer Zone Discussion Forum: Does ESP8266 actually have hardware I2C?". Espressif Systems. 2014-10-27. Retrieved 2017-10-02.
24. <https://www.etechnophiles.com/nodemcu-esp8266-pinout-specs-board-layout/>
25. <https://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/>
26. <https://www.instructables.com/ESP8266-Pro-Tips/>
27. <https://techzeero.com/sensors-modules/esp8266-wifi-module/>
28. <https://electropeak.com/learn/interfacing-dht21-am2301-temperature-humidity-sensor-with-arduino/>
29. https://www.espressif.com/sites/default/files/documentation/0aesp8266ex_datasheet_en.pdf.
30. <https://protosupplies.com/product/lcd2004-20x4-blue-lcd-display/>
31. <https://artofcircuits.com/product/white-on-blue-20x4-lcd-module-5v>
32. <https://www.thethingsindustries.com/docs/integrations/cloud-integrations/ubidots/>
33. <https://how2electronics.com/iot-ecg-monitoring-ad8232-ecg-sensor-esp8266/>
34. <https://randomnerdtutorials.com/esp8266-nodemcu-send-email-smtp-server-arduino/>
35. <https://www.hopkinsmedicine.org/health/conditions-and-diseases/vital-signs-body-temperature-pulse-rate-respiration-rate-blood-pressure>.
36. MORAN, Daniel S.; MENDAL, Liran. Core temperature measurement. *Sports Medicine*, 2002, 32.14: 879-885.
37. <https://www.nbcnews.com/health/health-news/it-s-not-just-heat-it-really-humidity-know-risks-n629486>
38. https://www.labfront.com/blog/hrv-in-research?utm_term=heart%20rate&utm_campaign=Blog:+HRV+in+Research&utm_source=adwords&utm_medium=ppc&hsa_acc=9515928370&hsa_cam=15772439494&hsa_grp=132323374512&hsa_ad=572866927118&hsa_src=g&hsa_tgt=kwd-10033811&hsa_kw=heart%20rate&hsa_mt=b&hsa_net=adwords&hsa_ver=3&gclid=CjwKCAjwoduRBhA4EiwACL5RP6w2QEu8b1bZKzQMdU5GKPXcQlccBljeb30VKTaZTDN902q0CyWY6xoCf7wQAvD_BwE
39. <https://www.bhf.org.uk/information-support/heart-matters-magazine/medical/ask-the-experts/pulse-rate>
40. <https://www.mayoclinic.org/healthy-lifestyle/fitness/expert-answers/heart-rate/faq-20057979>
41. <https://www.childrens.com/health-wellness/is-your-childs-heart-rate-healthy>
42. <https://www.uptodate.com/contents/image/print?imageKey=EM%2F78097>
43. G De Simone, RB Devereux, SR Daniels, RA Meyer. Gender differences in left ventricular growth. *Hypertension*. 1995;26:979–83.
44. R Zak. Development and proliferative capacity of cardiac muscle cells. *Circ Res*. 1974;35(II):17–26.
45. *European Heart Journal*; Heart Rate Variability and Heart Rate in Healthy Volunteers; D. Ramakers, et al. *J Oral Med*. 1998
46. AM Grandi, A Venco, F Barzizza, F Scalise, P Pantaleo, G Finardi. Influence of age and sex on left ventricular anatomy and function in normals. *Cardiology*. 1992;81:8–13.
47. G Olivetti, G Giordano, D Corradi, M Melissari, C Lagrasta, et al. Gender differences and ageing: effects on human heart. *J Am Coll Cardiol*. 1995;26:1068–79.

48. MJ Sullivan, FR Cobb, MB Higginbotham. Stroke volume increases by similar mechanisms during upright exercise in normal men and women. *Am J Cardiol.* 1991;67:1405–12.
49. MB Higginbotham, KG Morris, RE Coleman, FR Cobb. Sex related differences in the normal cardiac response to upright exercise. . *Circulation.* 1984;70:357–66.
50. A Deswal, B Bozkurt. Comparison of morbidity in a women versus men with heart failure and preserved ejection fraction . *Am J Cardiol.* 2006;97:1228–31.
51. V Vaccarino, L Parsons, NR Every, HV Barron, HM Krumholz. Sex based differences in early mortality after myocardial infarction. National Registry of Myocardial infarction 2 Participants. *N England J med.* 1999;341:217–25.
52. A Hofman, DE Grobbee, PT de Jong, FA Van Den Ouweland. Determinants of disease and disability in the elderly : the Rotterdam elderly study. *Eur JEpidemiol.* 1991;7:403–22.
53. PS Douglas, SE Kartz, EO Weinberg, MH Chen, SP Bishop, BH Lorell. Hypertrophic remodelling: gender differences in the early response of left ventricular pressure overload. *J Am Coll cardiol.* 1998;32:1118–25.
54. H Ni. Prevalence of self reported heart failure among US adults: results from the 1999 National Health Interview survey. *Am heart J.* 2003;146:121–28.
55. J Lindfeld, H Krause-Steinrauf, J Salerno. Where are all the women with heart failure? *Am Coll cardiol.* 1997; 30:1417–19.
56. SB Mohan, M Parker, M Wehbi, P Douglass. Idiopathic dilated cardiomyopathy: a common but mystifying cause of heart failure. *Cleve. Clin J Med.* 2002;69:481–87.
57. J Fernandez-Sola, JM Nicolas- Arfelis, D Corradi, M Melissari, C Lagrasta, et al. Gender differences in alcoholic cardiomyopathy. . *J Gend Specif Med.* . 2002;5:41–47.
58. <https://www.engineersgarage.com/arduino-based-ecg-cardiac-monitor-ad8232/>