BÁO CÁO ĐỒ ÁN

**Môn học: Công nghệ Internet of things hiện đại**

**Tên đề tài: Hệ thống dự đoán mưa**

*GVHD: Nguyễn Khánh Thuật*

1. **THÔNG TIN CHUNG:**

Lớp: NT532.O21

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Nguyễn Quang Thắng | 21522591 | 21522591@gm.uit.edu.vn |
| 2 | Bùi Quốc Huy | 21520911 | 21520911@gm.uit.edu.vn |
| 3 | Nguyễn Vĩnh Thái | 20520755 | 20520755@gm.uit.edu.vn |

1. **PHÂN CÔNG CÔNG VIỆC:**

|  |  |
| --- | --- |
| **Công việc** | **Thành viên thực hiện/đóng góp** |
| Lên ý tưởng về đề tài | Cả nhóm |
| Lên kế hoạch thực hiện đề tài | Huy |
| Tìm kiếm và thu thập datasets | Thắng |
| Thu mua, chuẩn bị phần cứng (cảm biến, raspberry, …) | Huy |
| Huấn luyện và đóng gói mô hình dự đoán | Huy |
| Xây dựng mã nguồn hoạt động của hệ thống | Thái |
| Thiết kế, cấu hình và triển khai hệ thống (phần cứng + phần mềm) | Huy |
| Viết báo cáo cuối kỳ | Thắng |
| Làm slide thuyết trình | Thái |
| Báo cáo cuối kỳ | Cả nhóm |

***Tổng kết đóng góp của các thành viên:***

* **Bùi Quốc Huy**: 60%
* **Nguyễn Quang Thắng**: 20%
* **Nguyễn Vĩnh Thái**: 20%

# **MỤC LỤC**

[**MỤC LỤC** 1](#_Toc169136958)

[**DANH MỤC HÌNH ẢNH** 2](#_Toc169136959)

[**DANH MỤC VIẾT TẮT** 3](#_Toc169136960)

[**CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI** 4](#_Toc169136961)

[***1.*** ***Lý do chọn đề tài*** 4](#_Toc169136962)

[***2.*** ***Tính cấp thiết của vấn đề nghiên cứu*** 4](#_Toc169136963)

[***3.*** ***Mục tiêu nghiên cứu*** 4](#_Toc169136964)

[***4.*** ***Phương pháp nghiên cứu*** 4](#_Toc169136965)

[**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT** 6](#_Toc169136966)

[***1.*** ***Raspberry Pi 4*** 6](#_Toc169136967)

[***2.*** ***Cảm biến nhiệt độ, độ ẩm, áp suất không khí BME280*** 7](#_Toc169136968)

[***3.*** ***Cảm biến mưa*** 8](#_Toc169136969)

[***4.*** ***Giao thức I2C*** 9](#_Toc169136970)

[***5.*** ***Google Colab*** 9](#_Toc169136971)

[***6.*** ***MLPClassifier*** 10](#_Toc169136972)

[***7.*** ***Blynk*** 10](#_Toc169136973)

[**CHƯƠNG 3. THIẾT KẾ, XẦY DỰNG KIẾN TRÚC HỆ THỐNG** 12](#_Toc169136974)

[***1.*** ***Tổng quan về hệ thống*** 12](#_Toc169136975)

[***2.*** ***Thiết kế hệ thống*** 13](#_Toc169136976)

[**CHƯƠNG 4. NỘI DUNG TRIỂN KHAI** 17](#_Toc169136977)

[***1.*** ***Trước khi triển khai*** 17](#_Toc169136978)

[***2.*** ***Triển khai*** 17](#_Toc169136979)

[***3.*** ***Kết quả*** 18](#_Toc169136980)

[**CHƯƠNG 5. KẾT LUẬN** 21](#_Toc169136981)

[***1.*** ***Kết luận chung*** 21](#_Toc169136982)

[***2.*** ***Ưu điểm*** 21](#_Toc169136983)

[***3.*** ***Nhược điểm*** 21](#_Toc169136984)

[**CHƯƠNG 6. HƯỚNG PHÁT TRIỂN** 22](#_Toc169136985)

[**TÀI LIỆU THAM KHẢO** 23](#_Toc169136986)

[**TRẢ LỜI CÂU HỎI KHI TRÌNH BÀY** 24](#_Toc169136987)

# **DANH MỤC HÌNH ẢNH**

[Hình 1: Raspberry Pi 4 6](#_Toc169136939)

[Hình 2: Giải nghĩa các chân Raspberry Pi 7](#_Toc169136940)

[Hình 3: BME280 8](#_Toc169136941)

[Hình 4: Cảm biến mưa 9](#_Toc169136942)

[Hình 5: Icon google colab 10](#_Toc169136943)

[Hình 6: Icon Blynk 11](#_Toc169136944)

[Hình 7: Tổng quan hệ thống 12](#_Toc169136945)

[Hình 8: Mô hình nối dây 13](#_Toc169136946)

[Hình 9: Thông tin về dataset 14](#_Toc169136947)

[Hình 10: Thông tin về tập dữ liệu sau các bước tiền xử lý 15](#_Toc169136948)

[Hình 11: Nhãn dùng để huấn luyện 15](#_Toc169136949)

[Hình 12: Các tham số cho MLPClassifier 15](#_Toc169136950)

[Hình 13: Kết quả huấn luyện 15](#_Toc169136951)

[Hình 14: Toàn cảnh mô hình trong thực tế 17](#_Toc169136952)

[Hình 15: Cận cảnh 2 cảm biến: BME280 và WRS 18](#_Toc169136953)

[Hình 16: Mail báo đang mưa đến 21520911@gm.ui.edu.vn 18](#_Toc169136954)

[Hình 17: Mail báo kết quả dự đoán 19](#_Toc169136955)

[Hình 18: Giao diện blynk trên web 19](#_Toc169136956)

[Hình 19: Giao diện blynk trên di động 20](#_Toc169136957)

# **DANH MỤC VIẾT TẮT**

|  |  |
| --- | --- |
| **Viết tắt** | **Viết đầy đủ** |
| IoT | Internet of Thing |
| SoC | System on a Chip |
| ATM | Automated Teller Machine |
| CPU | Central Processing Unit |
| GPU | Graphics Processing Unit |
| RAM | Random Access Memory |
| SD | Secure Digital |
| USB | Universal Serial Bus |
| UDP | User Datagram Protocol |
| CLR | Common Language Runtime |
| VCC | Voltage at the Common Collector |
| GND | Ground |
| SCL | Serial Clock |
| SDA | Serial Data |
| ADDR | Address |
| I2C | Inter-Integrated Circuit |
| UART | Universal Asynchronous Receiver/Transmitter |
| SPI | Serial Peripheral Interface |
| WRS | Water Rain Sensor |

# **CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI**

1. ***Lý do chọn đề tài***

Dự án xây dựng hệ thống dự đoán mưa được chúng em lựa chọn vì nhu cầu ngày càng tăng của các ngành công nghiệp, nông nghiệp và đời sống hàng ngày. Việc có thể dự đoán mưa chính xác sẽ giúp giảm thiểu rủi ro, tối ưu hóa kế hoạch sản xuất, cung cấp và vận chuyển, cũng như tăng cường sự an toàn cho các hoạt động ngoài trời.

1. ***Tính cấp thiết của vấn đề nghiên cứu***

Tính cấp thiết của đề tài này là không thể phủ nhận, đặc biệt trong bối cảnh biến đổi khí hậu đang diễn ra nhanh chóng. Việc nắm bắt được dự báo chính xác về mưa sẽ giúp chúng ta phản ứng kịp thời và hiệu quả trước các tác động tiềm ẩn của thời tiết.

1. ***Mục tiêu nghiên cứu***

Mục tiêu của nghiên cứu là phát triển một hệ thống iot đưa ra các dự đoán mưa dựa trên dữ liệu môi trường và ngày trong năm có độ chính xác cao và khả năng cập nhật linh hoạt. Mô hình dự đoán mưa dựa trên dữ liệu môi trường sẽ sử dụng dữ liệu từ các cảm biến cơ bản để đưa ra dự đoán, được huấn luyện dựa vào một datasets về mối liên hệ giữa dữ liệu môi trường và mưa. Mô hình dự đoán mưa dựa trên ngày trong năm sẽ được huấn luyện dựa vào dữ liệu mưa địa phương trong nhiều năm.

Hệ thống sẽ gửi các kết quả dự đoán đến người dùng qua email.

Hệ thống có thu thập các mẫu dữ liệu cho các cuộc huấn luyện mô hình trong tương lai.

1. ***Phương pháp nghiên cứu***

Phương pháp nghiên cứu sẽ bao gồm việc áp dụng các thuật toán máy học và học sâu để phân tích và dự đoán mẫu thời tiết.

Phương pháp tìm kiến thức: Internet, chatGPT, tài liệu Raspberry,…

Ngôn ngữ lập trình:

* Python3

Nền tảng huấn luyện mô hình học máy: Google Colab

# **CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

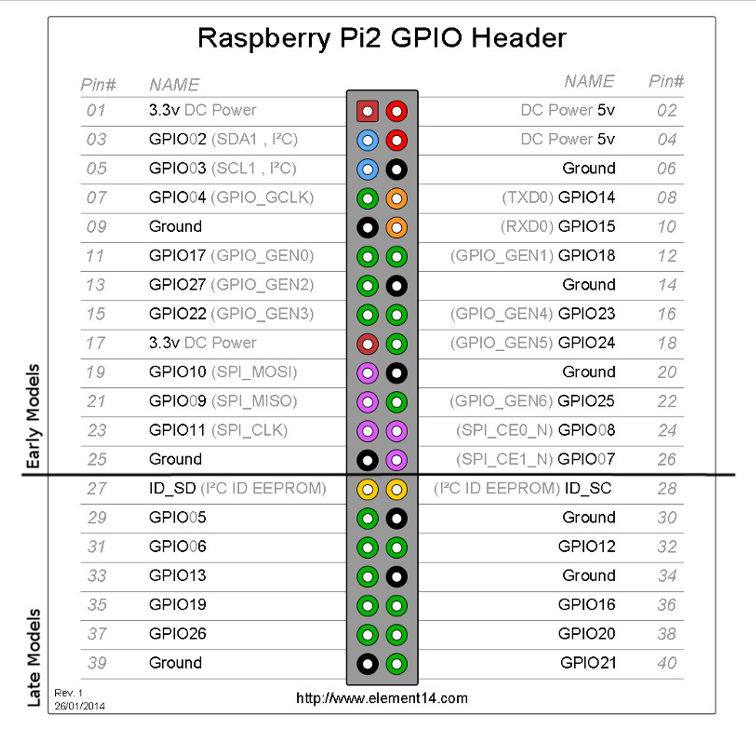
1. ***Raspberry Pi 4***

Raspberry Pi là một máy vi tính rất nhỏ gọn, kích thước hai cạnh chỉ cỡ một cái thẻ ATM, trong đó đã tích hợp mọi thứ cần thiết để bạn sử dụng như một máy vi tính. Bộ xử lý SoC Broadcom BCM2835 của nó bao gồm CPU, GPU, RAM, khe cắm thẻ microSD, WiFi, Bluetooth và 4 cổng USB 2.0.



Hình 1: Raspberry Pi 4

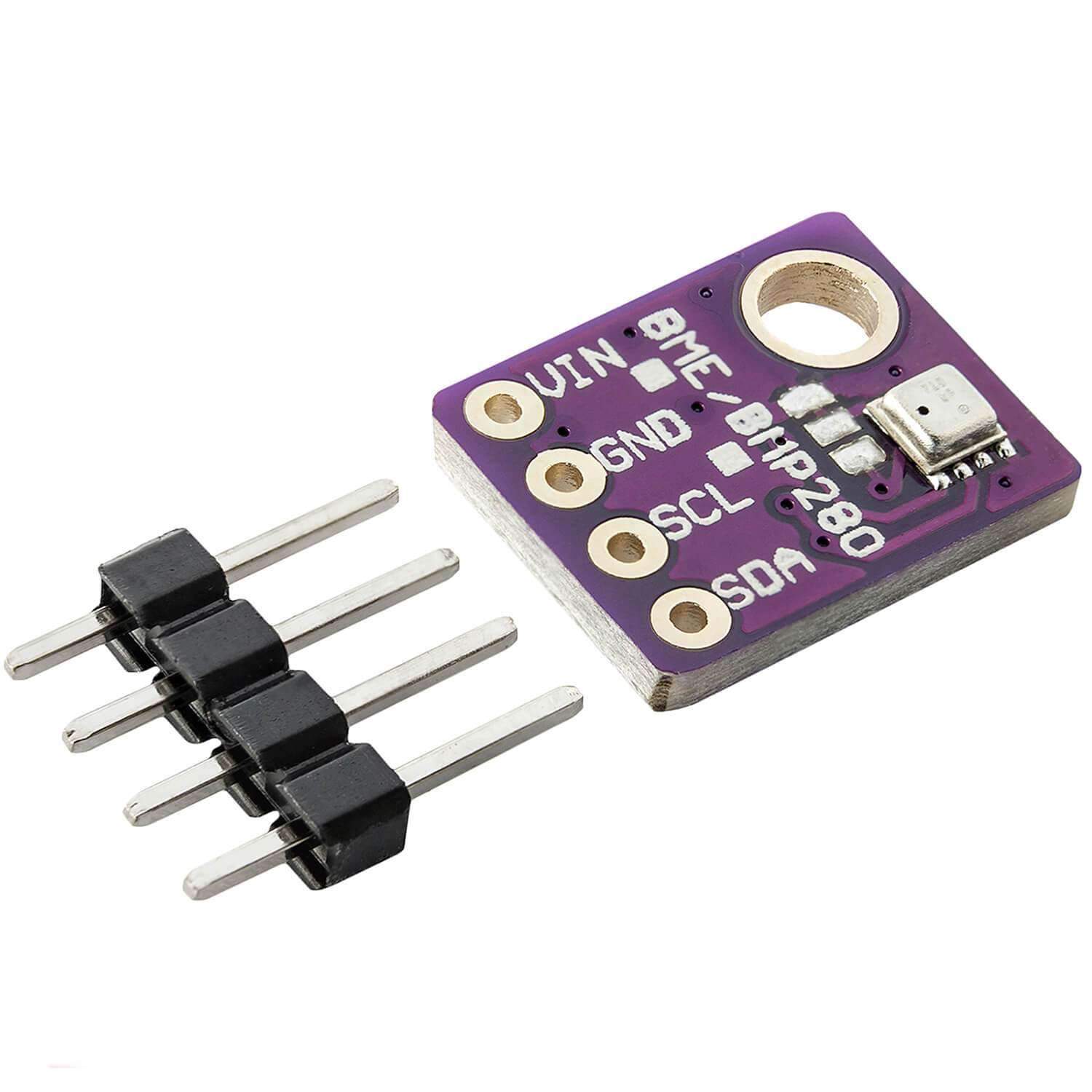
Giải nghĩa các chân của Raspberry Pi 4:



Hình 2: Giải nghĩa các chân Raspberry Pi

1. ***Cảm biến nhiệt độ, độ ẩm, áp suất không khí BME280***

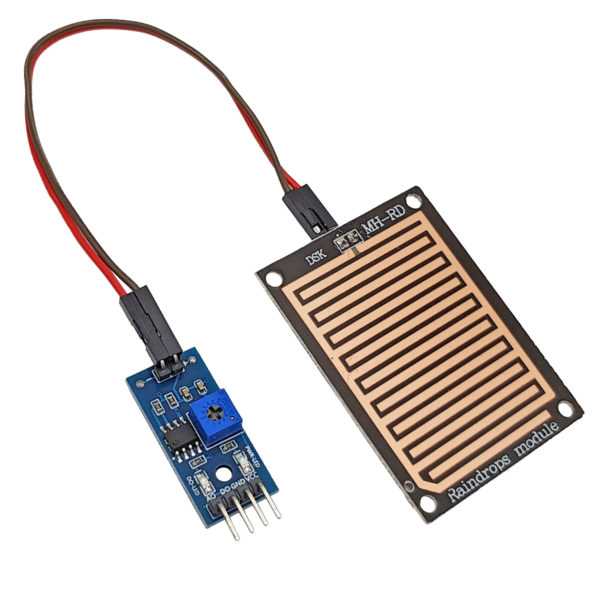
* Phạm vi đo độ ẩm: 0 ~ 100%
* Phạm vi đo nhiệt độ: -40…85°C
* Phạm vị đo áp suất: 300...1100 hPa
* Giao thức: I²C và SPI
* Cung cấp điện áp VDD: 1,71V đến 3,6V
* Cung cấp điện áp VDDIO: 1,2V đến 3,6V
* Tiêu thụ điện trung bình (tiêu chuẩn) (tỷ lệ làm mới dữ liệu 1Hz):
  + 1.8 μA @ 1 Hz (H, T)
  + 2.8 μA @ 1 Hz (P, T)
  + 3.6 μA @ 1 Hz (H, P, T)
  + T = temperature
* Tiêu thụ điện trung bình ở chế độ ngủ: 0.1 μA



Hình 3: BME280

1. ***Cảm biến mưa***

* Điện áp: 5V
* Led báo nguồn (màu xanh), Led cảnh báo mưa (màu đỏ)
* Hoạt động dựa trên nguyên lý: Nước rơi vào board sẽ tạo ra môi trường dẫn điện.
* Có 2 dạng tín hiệu: Analog (AO) và Digital (DO)
* Dạng tín hiệu : TTL, đầu ra 100mA (có thể sử dụng trực tiếp Relay, còi công suất nhỏ…)
* Điều chỉnh độ nhạy bằng biến trở.
* Sử dụng LM358 để chuyển AO –> DO



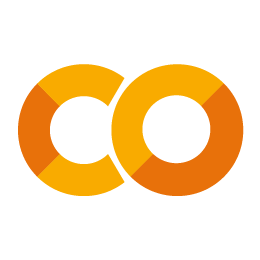
Hình 4: Cảm biến mưa

1. ***Giao thức I2C***

* I2C kết hợp các tính năng tốt nhất của SPI và UART. Với I2C, bạn có thể kết nối nhiều slave với một master duy nhất (như SPI) và bạn có thể có nhiều master điều khiển một hoặc nhiều slave.
  + UART hay bộ thu-phát không đồng bộ đa năng là một trong những hình thức giao tiếp kỹ thuật số giữa thiết bị với thiết bị đơn giản và lâu đời nhất.
  + SPI – Serial Peripheral Interface là giao tiếp đồng bộ, bất cứ quá trình nào cũng đều được đồng bộ với xung clock sinh ra bởi thiết bị Master.
* I2C chỉ sử dụng hai dây để truyền dữ liệu giữa các thiết bị:
  + SDA (Serial Data) - đường truyền cho master và slave để gửi và nhận dữ liệu.
  + SCL (Serial Clock) - đường mang tín hiệu xung nhịp.

1. ***Google Colab***

* Google Colab, viết tắt của Google Colaboratory, là một dịch vụ cung cấp môi trường Jupyter Notebook hoàn toàn trực tuyến. Nó cho phép người dùng tạo, chia sẻ và chỉnh sửa các tệp notebook một cách dễ dàng mà không cần cài đặt bất kỳ phần mềm nào.
* Mục đích chính của Google Colab là hỗ trợ người dùng trong việc phát triển và chia sẻ các dự án liên quan đến khoa học dữ liệu, học máy (machine learning), và nghiên cứu khoa học thông qua môi trường lập trình Python dễ sử dụng.



Hình 5: Icon google colab

1. ***MLPClassifier***

* MLPClassifier là một lớp học trong thư viện Scikit-learn được dùng để đào tạo mạng nơ-ron đa lớp (MLP) cho các nhiệm vụ phân loại. MLP là một loại mạng nơ-ron hướng tiếp (feedforward) bao gồm nhiều lớp nơ-ron, mỗi lớp được kết nối với lớp tiếp theo.
* Lớp MLPClassifier cung cấp nhiều tùy chọn để xác định kiến trúc mạng, đào tạo mô hình và đánh giá hiệu suất của nó. Nó có thể được sử dụng cho nhiều nhiệm vụ phân loại khác nhau, bao gồm phân loại hình ảnh, phân loại văn bản và phân loại chuỗi thời gian.

1. ***Blynk***

* Blynk là một nền tảng IoT (Internet of Things) giúp người dùng dễ dàng tạo ra các ứng dụng điều khiển từ xa cho các thiết bị thông minh. Điểm đặc biệt của Blynk chính là sự linh hoạt và dễ sử dụng.



Hình 6: Icon Blynk

# **CHƯƠNG 3****. THIẾT KẾ, XẦY DỰNG KIẾN TRÚC HỆ THỐNG**

1. ***Tổng quan về hệ thống***

**Các thành phần:**

- Raspberry pi 4

- Cảm biến nhiệt nhiệt độ, độ ẩm, áp suất không khí BME280

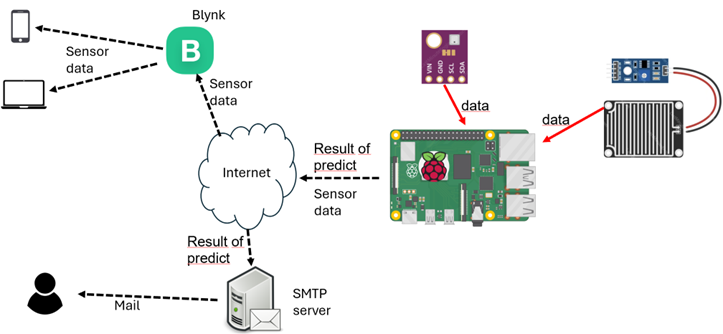
- Cảm biến mưa

- Breadboard

- Model máy học dự đoán mưa dựa trên dữ liệu môi trường (rain\_predict\_model.joblib)

**Tổng quan thiết kế hệ thống:**

* Từ dữ liệu nhận được từ các cảm biến, raspberry dùng dữ liệu đó và model để dự đoán xem ngày tiếp theo có mưa hay không và gửi email thông báo kết quả đến các subscribers. Dữ liệu từ ngày hôm trước sẽ được sử dụng và email sẽ được gửi lúc 00:00 ngày hôm sau nên thông điệp sẽ là “… Today will (not) rain”.
* Nếu có mưa sẽ gửi mail cảnh báo đến các subscribers. Thông điệp là “… It is raining here”.
* Các số liệu cảm biến nhận được sẽ được cập nhật thường xuyên lên Blynk.
* Các dữ liệu thu thập được cũng sẽ được lưu để có thể train model mới trong tương lai.



Hình 7: Tổng quan hệ thống

1. ***Thiết kế hệ thống***

* Cách nối dây:
  + SCL và SDA của BME280 nối với chân 5 và 3 của raspberry.
  + D0 của Rain Water Sensor nối với chân 11 của raspberry.

A diagram of a circuit board

Description automatically generated

Hình 8: Mô hình nối dây

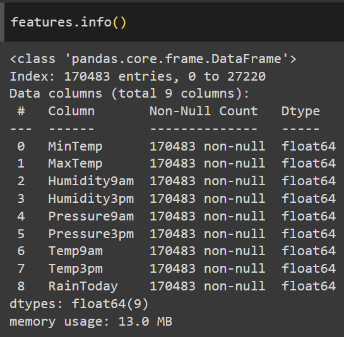
* Train model dự đoán mưa:
  + Dataset: Dữ liệu thời tiết hàng ngày trong 10 năm từ các trạm thời tiết ở Úc. Nguồn dataset: [weatherAUS.csv](https://www.kaggle.com/code/chandrimad31/rainfall-prediction-7-popular-models/input)
    - Tổng 145460 mẫu.
    - Tất cả 22 đặc trưng: Date, Location, MinTemp, MaxTemp, Rainfall, Evaporation, Sunshine, WindGustDir, WindGustSpeed, WindDir9am, WindDir3pm, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, RainToday (yes: rain/no: not rain**)**.
    - Nhãn: RainTomorrow (yes: rain/no: not rain**).**

A screenshot of a computer

Description automatically generated

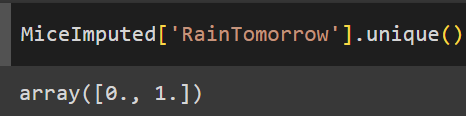
Hình 9: Thông tin về dataset

* + Notebook train model: xem tệp Notebook train model.ipynb hoặc [nt532-rainfall\_next\_day\_predict.ipynb](https://colab.research.google.com/drive/1xSMal8_Jr6dx1hbf_w_nKaVA8GS1zqU0?hl=vi#scrollTo=M14AQXPQ1W-C)
    - Thông qua các bước tiền xử lý dữ liệu (cân bằng dữ liệu, loại bỏ nhiễu, lựa chọn đặc trưng), thì tổng sử dụng 170483 mẫu, chia 75 – 25 để huấn luyện và kiểm tra.
    - Sử dụng 9 đặc trưng: MinTemp, MaxTemp, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Temp9am, Temp3pm, RainToday (1: rain/0: not rain**)**.



Hình 10: Thông tin về tập dữ liệu sau các bước tiền xử lý

* + - Nhãn: RainTomorrow (1: rain/0: not rain**).**



Hình 11: Nhãn dùng để huấn luyện

* + - Sử dụng trình phân loại: MLPClassifier

A screen shot of a computer

Description automatically generated

Hình 12: Các tham số cho MLPClassifier

* + - Độ chính xác của mô hình: ~73,5%

A screenshot of a computer

Description automatically generated

Hình 13: Kết quả huấn luyện

* + Tham khảo: dự án của Chandrima D - [Rainfall Prediction: 7 Popular Models](https://www.kaggle.com/code/chandrimad31/rainfall-prediction-7-popular-models/notebook/).
* Code hoạt động của hệ thống trên Raspberry: xem tệp WeatherStation.py
  + Cách dùng: ./WeatherStation.py subscriber1 subscriber2 …
* Chạy code bằng lệnh nohup để chương trình không bị ngắt khi ngắt phiên ssh: nohup ./WeatherStation.py [21520911@gm.uit.edu.vn](mailto:21520911@gm.uit.edu.vn) > output.log & . Trong đó [21520911@gm.uit.edu.vn](mailto:21520911@gm.uit.edu.vn) là subscriber, output.log để lưu output của chương trình thường là các warning và error, & để chạy ở background.

# **CHƯƠNG 4. NỘI DUNG TRIỂN KHAI**

1. ***Trước khi triển khai***
2. **Chuẩn bị dữ liệu**

* File rain\_predict\_model.joblib lưu model dự đoán.
* Các email để làm subscriber.

1. **Chuẩn bị môi trường**

* Nơi triển khai cần gần nguồn điện cho raspberry và an toán về mặt vật lý cho hệ thống. Nên nới triển khai là phòng ktx của Bùi Quốc Huy.

1. **Kịch bản**

* Kịch bản 1: Khi có mưa thì có email thông báo có mưa đến subscribers.
* Kịch bản 2: Khi chuyển ngày gửi email thông báo kết quả dự đoán mưa của ngày hôm sau.
* Kịch bản 3: Lưu lại dữ liệu cho lượt train model sau.
* Kịch bản 4: Dữ liệu của các cảm biến được cập nhật thường xuyên lên Blynk.

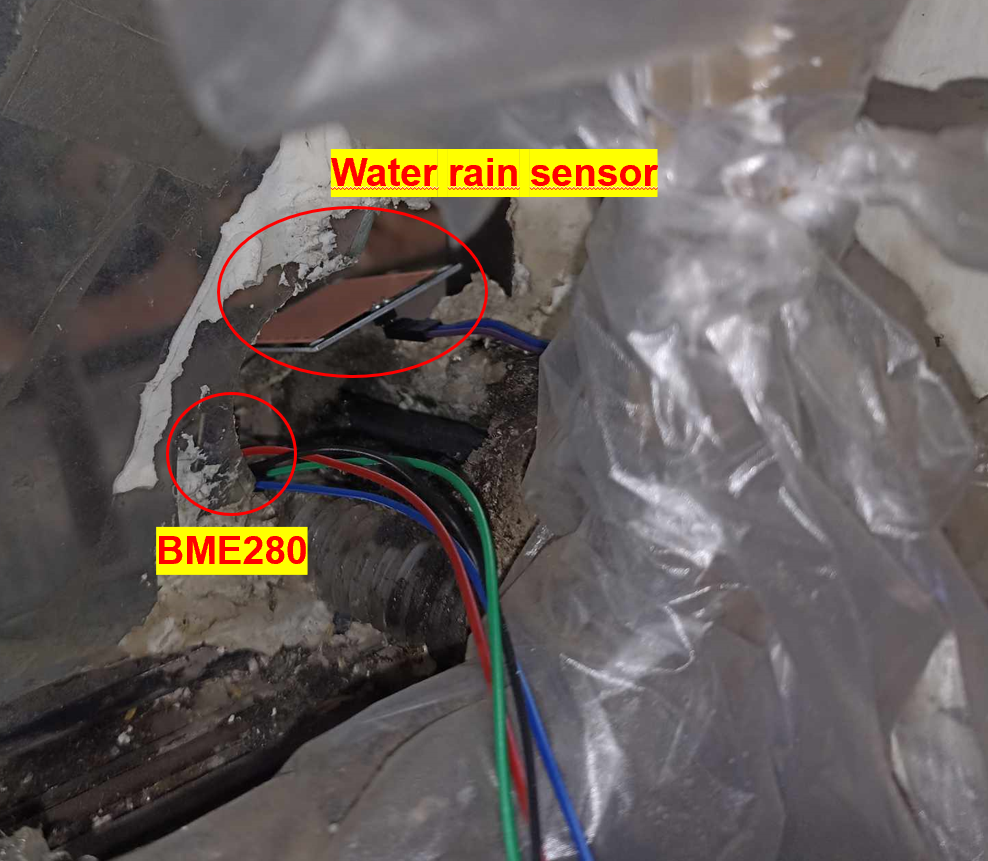
1. ***Triển khai***

* Hệ thống được triển khai từ ngày 5/5 nhưng do nhóm code không kỹ nên đã phải sửa code liên tục đến 9/5 thì phải dừng và không thu được nhiều thành quả.
* Các minh chứng: video [vid\_mc.mp4](https://drive.google.com/file/d/1JDiY4OsGLyOX6KWwHZ98b6ymUWSbmSmX/view?usp=drive_link) và vài ảnh.

A close-up of several wires

Description automatically generated

Hình 14: Toàn cảnh mô hình trong thực tế



Hình 15: Cận cảnh 2 cảm biến: BME280 và WRS

1. ***Kết quả***

* **Kịch bản 1**: Kết quả đúng như mong đợi, khi có mưa có mail gửi về cho người dùng báo có mưa.

A screenshot of a computer

Description automatically generated

Hình 16: Mail báo đang mưa đến 21520911@gm.ui.edu.vn

* **Kịch bản 2**: Kết quả đúng như mọng đợi, kết quả dự đoán được gửi qua mail cho người dùng. Tuy nhiên đã gửi sai giờ, do sai Time Zone.

**A screenshot of a computer

Description automatically generated**

Hình 17: Mail báo kết quả dự đoán

* **Kịch bản 3**: Thất bại, nhóm không xuất ra được bất kỳ tập dữ liệu nào, do cần ít nhất 2 ngày liên tục hoạt động bình thường để thu được 1 mẫu nhưng do code lỗi liên tục nên không thể thu được 1 mẫu nào.
* **Kịch bản 4**: Kết quả như mong đợi, người dùng có thể theo dõi dữ liệu môi trường mà cảm biến thu được từ xa qua giao diện của Blynk. Nhưng, đôi lúc raspberry bị mất kết nối với Blynk không rõ nguyên do.

**A screenshot of a computer

Description automatically generated**

Hình 18: Giao diện blynk trên web

A screenshot of a device

Description automatically generated

Hình 19: Giao diện blynk trên di động

# **CHƯƠNG 5. KẾT LUẬN**

1. ***Kết luận chung***

* Nhóm đã xây dựng được 1 hệ thống IoT cơ bản có thể đưa ra các dự đoán mưa dựa trên dữ liệu môi trường.
* Nhóm đã huấn luyện được một mô hình dự đoán mưa dựa trên dữ liệu mô trường.
* Nhóm đã không huấn luyện được mô hình dự đoán mưa dựa trên ngày trong năm do không tìm thấy dataset về mưa của khu vực địa phương.

1. ***Ưu điểm***

* Người dùng có thể theo dõi dữ liệu môi trường thông qua dữ liệu cảm biến được cập nhật lên blynk.
* Người dùng có thể nhận được các mail kết quả dự đoán mưa trong ngày để chuẩn bị.

1. ***Nhược điểm***

* Hệ thống chưa hoàn thiện, vẫn chưa thu thập được mẫu, code vẫn còn bug.
* Thời gian triển khai quá ngắn không thấy được hiệu quả lâu dài cũng như không đủ thời gian hoàn thiện hệ thống.
* Thiếu chức năng dự đoán mưa theo ngày trong năm mà nhóm đã đề ra ban đầu.
* Dữ liệu môi trường dùng để dự đoán chưa đủ chi tiết để dự đoán mưa, nên thêm các dữ liệu về tốc độ gió, hướng gió, nắng, …

# **CHƯƠNG 6. HƯỚNG PHÁT TRIỂN**

* Cải thiện code để không còn Bug.
* Thêm dữ liệu tốc độ gió, hướng gió, nắng, lượng mưa, … theo dataset gốc cho tập huấn luyện và huấn luyện lại mô hình cũng như thêm các cảm biến liên quan như cảm biến tốc độ gió, hướng gió, nắng, lượng mưa, … cho hệ thống.
* Thêm chức năng dự đoán mưa theo ngày bằng mô hình máy học hoặc lấy kết quả từ một web thời tiết nào đó có dự đoán mưa trên khu vực.
* Nghiên cứu mở rộng dự đoán lượng mưa.

# **TÀI LIỆU THAM KHẢO**

* Lý thuyết về I2C: <https://dientutuonglai.com/chuan-giao-tiep-i2c-la-gi.html>
* Lý thuyết về SPI: <https://deviot.vn/blog/giao-tiep-spi.74706311>
* Lý thuyết về UART: <https://dientutuonglai.com/giao-tiep-uart-la-gi.html>
* Lý thuyết về Raspberry Pi 4: [Raspberry Pi 4 là gì?](https://mecsu.vn/ho-tro-ky-thuat/raspberry-pi-4-la-gi-so-do-chan-tinh-nang-va-ngoai-vi-cua.lmA#:~:text=Raspberry%20pi%204%20l%C3%A0%20m%E1%BB%99t,hi%E1%BB%87u%20%C4%91%E1%BA%A7u%20ra%20nhanh%20ch%C3%B3ng)
* Lý thuyết về BME280: [BME280](https://www.bosch-sensortec.com/media/boschsensortec/downloads/product_flyer/bst-bme280-fl000.pdf)
* Lý thuyết về cảm biến mưa: <https://nshopvn.com/product/cam-bien-mua/>
* Lý thuyết về Google Colab: [Google Colab là gì?](https://www.matbao.net/tin-tuc/google-colab-la-gi-kham-pha-nen-tang-so-ghi-chep-tinh-toan-truc-tuyen-134477.html)
* Lý thuyết về Blynk: [Blink là gì?](https://fptshop.com.vn/tin-tuc/danh-gia/blynk-la-gi-184300#:~:text=1.-,Blink%20l%C3%A0%20g%C3%AC%3F,ho%E1%BA%A1t%20v%C3%A0%20d%E1%BB%85%20s%E1%BB%AD%20d%E1%BB%A5ng.)
* Dự án tham khảo: [Rainfall Prediction: 7 Popular Models](https://www.kaggle.com/code/chandrimad31/rainfall-prediction-7-popular-models/notebook/)
* Dataset cho dự đoán mưa dựa trên dữ liệu môi trường: [weatherAUS.csv](https://www.kaggle.com/code/chandrimad31/rainfall-prediction-7-popular-models/input)l

# **TRẢ LỜI CÂU HỎI KHI TRÌNH BÀY**

Nhóm không nhớ và không có ghi chép về các câu hỏi cho nhóm khi trình bày.

Nhóm chân thành xin lỗi về bất cẩn của nhóm.