

TRƯỜNG THPT CHUYÊN LÊ QUÝ ĐÔN
CUỘC THI KHOA HỌC KỸ THUẬT CHO HỌC SINH TRUNG HỌC
NĂM HỌC 2016 - 2017

THIẾT KẾ HỆ THỐNG THEO DÕI VÀ CẢNH BÁO
CHẤT LƯỢNG KHÔNG KHÍ

Lĩnh vực dự thi

Khoa học kỹ thuật

Tác giả

Hồ Tuấn Kiệt

Lớp 11A5, Trường THPT chuyên Lê Quý Đôn, thành phố Đà Nẵng

MỤC LỤC

LỜI CẢM ƠN.....	3
TÓM TẮT NỘI DUNG DỰ ÁN.....	4
GIỚI THIỆU VÀ TỔNG QUAN VỀ VẤN ĐỀ NGHIÊN CỨU.....	5
GIẢ THUYẾT KHOA HỌC VÀ PHÁT BIỂU MỤC ĐÍCH NGHIÊN CỨU.....	5
PHƯƠNG PHÁP NGHIÊN CỨU.....	6
SỐ LIỆU/ KẾT QUẢ NGHIÊN CỨU.....	7
PHÂN TÍCH SỐ LIỆU/ KẾT QUẢ VÀ THẢO LUẬN.....	8
KẾT LUẬN.....	9
TÀI LIỆU THAM KHẢO.....	10

LỜI CẢM ƠN

TÓM TẮT NỘI DUNG DỰ ÁN

Sự công nghiệp hoá đã mang lại nhiều lợi ích cho con người, tuy nhiên kèm theo đó là tình trạng ô nhiễm môi trường càng ngày càng nghiêm trọng, trong đó đặc biệt nhất là tình trạng ô nhiễm không khí trầm trọng ở nhiều quốc gia. Vì vậy, việc theo dõi mức độ ô nhiễm không khí là một điều cần thiết để cảnh báo kịp thời cho người dân khi mức độ ô nhiễm vượt quá giới hạn cho phép.

Nhận thấy được điều đó, tôi đã thực hiện nghiên cứu một hệ thống có khả năng nhận biết mức độ ô nhiễm của không khí và từ đó có thể tự động đưa ra thông báo khi cần thiết.

Để đạt được các mục tiêu ở trên, hệ thống bao gồm các thành phần như sau:

- Một thiết bị được trang bị một số cảm biến về các loại khí trong không khí, có khả năng đo được nồng độ các loại khí trên.
- Một máy chủ (?) lưu trữ dữ liệu được truyền lên bởi các thiết bị quan trắc. Ngoài việc này máy chủ này còn có khả năng hiển thị thông tin dưới dạng trang web và đưa ra thông báo dưới dạng email và tin nhắn SMS khi cần thiết

Đóng góp mới của dự án này là một hệ thống theo dõi chất lượng không khí có giá thành rẻ, có thời gian triển khai ngắn và khả năng mở rộng nhanh chóng từ quy mô nhỏ cho tới quy mô lớn.

GIỚI THIỆU VÀ TỔNG QUAN VỀ VẤN ĐỀ NGHIÊN CỨU

Một trong những vấn đề đang được các nước quan tâm hiện nay chính là tình trạng ô nhiễm không khí trầm trọng đang xảy ra ở nhiều khu vực trên thế giới[citation needed], ví dụ như ở Trung Quốc[citation needed].

Theo [Kampa & Castanas 2008], các nhân tố có hại đối với sức khỏe con người trong không khí bao gồm:

- **Chất ô nhiễm dạng ga (Sox, NOx, CO, O3, ...)**
- **Các phần tử bụi**
- Kim loại nặng
- Persistent organic pollutants

Trong các nhân tố kể trên, xxx và xxx là những nhân tố được đề cập nhiều nhất khi nói tới ô nhiễm không khí, vì những nhân tố này được tạo ra từ những hoạt động thường gặp nhất ở con người, như hoạt động sản xuất công nghiệp, hoặc hoạt động sử dụng các phương tiện đốt nhiên liệu hóa thạch.

Đã có rất nhiều nghiên cứu được tiến hành để đánh giá ảnh hưởng của các nhân tố này đối với sức khỏe con người. Ví dụ như:

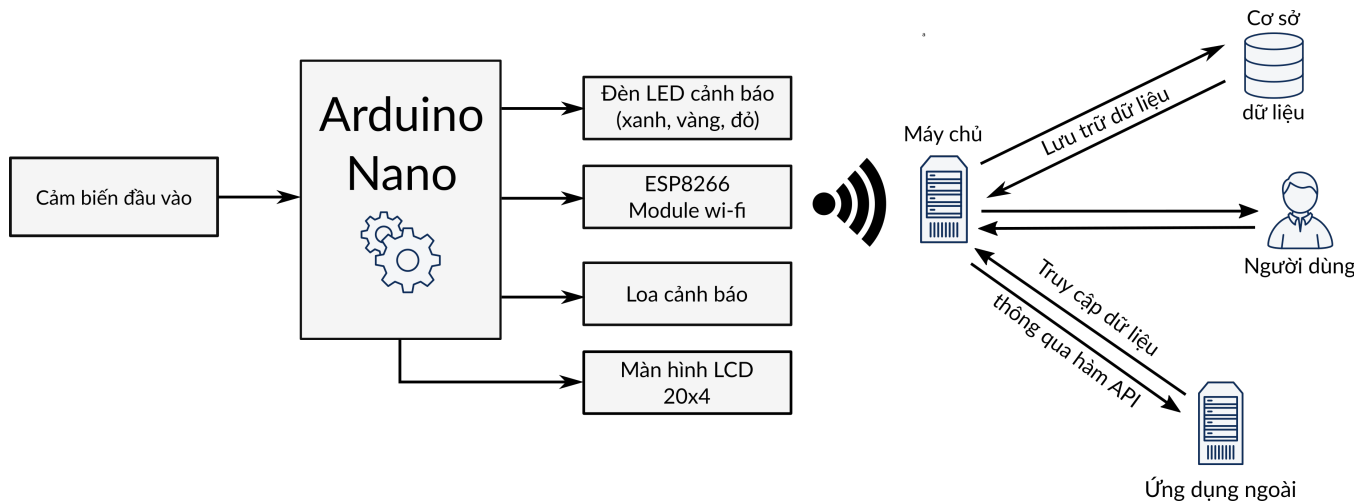
- Việc hít thở khí SO₂ ở nồng độ từ 0.5 tới 1.0 ppm trong thời gian ngắn (<3 phút) có thể gây co thắt phế quản [Balmes, Fine & Sheppard 1987]
- Nồng độ tăng dần lượng NO_x trong máu có liên quan tới việc gia tăng số lần nhập viện vì bệnh hen suyễn [Bernstein et al. 2004]
- xxx
- xxx
- xxx

Nghiên cứu được thực hiện dựa trên các cơ sở sau:

- Không khí là hỗn hợp của rất nhiều các loại khí khác nhau. Trong đó, có một số loại khí có thể gây ảnh hưởng tới sức khỏe con người như^{[1][2]}:
 - Khí SO_x, NO_x
 - Khí CO

- Khí ozone (mặc dù khí ozone là thành phần quan trọng của lớp ozone bảo vệ chúng ta khỏi tia cực tím của mặt trời, khí ozone trong bầu khí quyển lại là một chất khí có hại^[3])
- Các phần tử bụi ví dụ như PM2.5 (phần tử có đường kính nhỏ hơn 2.5 μm) và PM10 (phần tử có đường kính nhỏ hơn 10 μm)
- Dựa trên nồng độ đo được ở trên, chúng ta có thể tính toán được chỉ số AQI (Air quality index, chỉ số chất lượng không khí)^[4]
- Sử dụng các cảm biến có trên thị trường, ta có thể đo được nồng độ các khí ở trên, ví dụ như:
 - Cảm biến MQ-x có khả năng đo nồng độ khí CO^[6].
 - *Cảm biến Sharp GP2Y1010AU0F có khả năng đo nồng độ bụi trong không khí ở đơn vị mg/m^3* ^[5]

GIẢ THUYẾT KHOA HỌC VÀ PHÁT BIỂU MỤC ĐÍCH NGHIÊN CỨU



Nghiên cứu được chia làm hai phần: thiết kế thiết bị quan trắc và thiết kế trang web người dùng.

Về thiết bị quan trắc:

- Mạch điều khiển chính là mạch Arduino Nano sẽ sử dụng các cảm biến đầu vào để xác định nồng độ các chất trong không khí
- Nồng độ đo được sẽ được thông báo trên màn hình LCD, đồng thời được gửi lên máy chủ nếu có kết nối
- Nếu nồng độ vượt quá mức cho phép thì thiết bị sẽ đổi màu của đèn LED và phát loa như là một phương thức cảnh báo.

Về trang web:

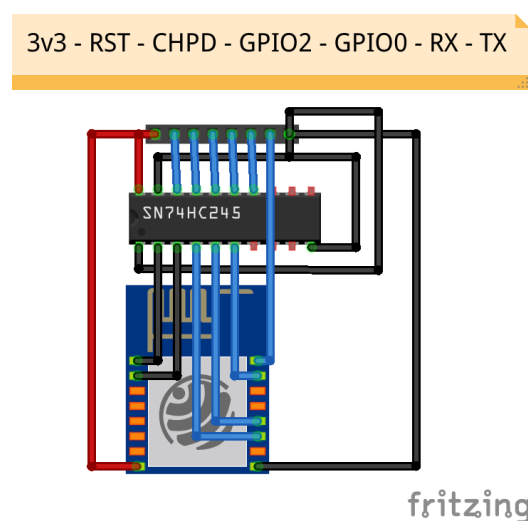
- Khi nhận được số liệu mới đăng, máy chủ sẽ lưu vào cơ sở dữ liệu. Nếu nồng độ vượt quá mức cho phép thì máy chủ sẽ tự động gửi email và tin nhắn thông báo tới những người đăng kí nhận tin từ thiết bị quan trắc đó.
- Khi truy cập vào trang web, người dùng có thể xem số liệu cũ của các trạm và theo dõi số liệu mới nhất của trạm đó.

PHƯƠNG PHÁP NGHIÊN CỨU

Thiết bị quan trắc

Thiết bị quan trắc được điều khiển bởi hai module chính:

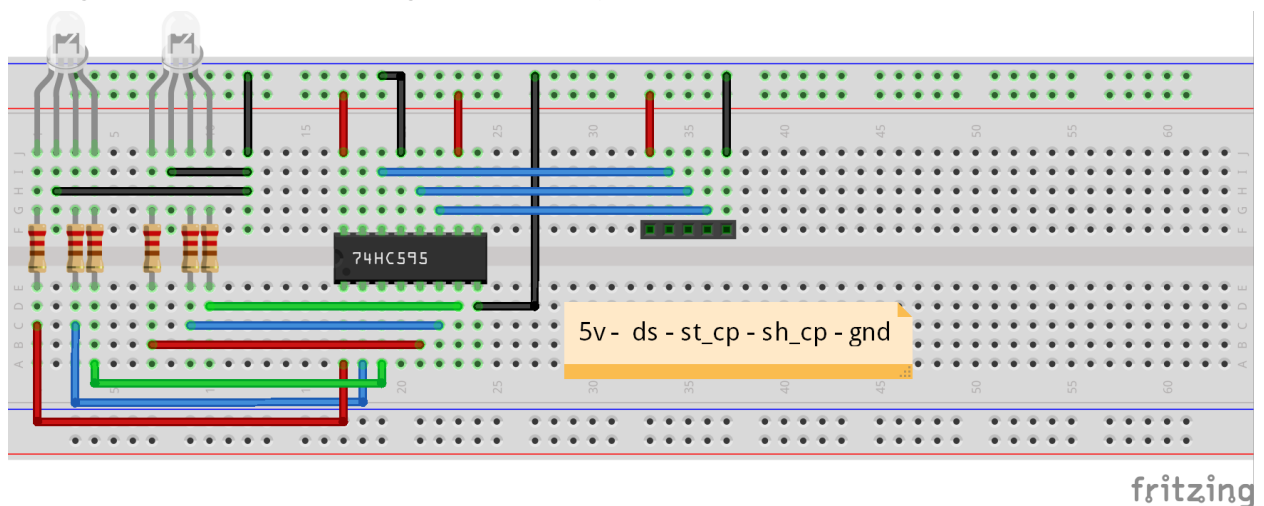
- Mạch Arduino Nano với vi điều khiển ATmega328P. Đây là mạch chính điều khiển hầu hết mọi hoạt động của thiết bị, bao gồm việc đọc số liệu từ các cảm biến, cảnh báo qua đèn và điều khiển màn hình LCD.
- Các hoạt động liên quan tới việc kết nối không dây được mạch ESP-12 đảm nhiệm. Mạch ESP-12 được trang bị chip ESP8266EX, tích hợp hầu hết các thành phần cần thiết như bộ thu phát RF, RAM, ROM và CPU trên một con chip nhỏ gọn. Mạch ESP-12 sẽ đảm nhiệm tất cả công việc có liên quan tới kết nối không dây, bao gồm việc gửi thông tin lên máy chủ.
- Vì mạch ESP-12 hoạt động ở hiệu điện thế 3.3v trong khi mạch Arduino Nano lại hoạt động ở hiệu điện thế 5v nên tín hiệu của mạch Arduino Nano bắt buộc phải đi qua IC 74HC245 để giảm hiệu điện thế tín hiệu từ 5v xuống 3.3v. IC 74HC245 là IC cho phép dẫn truyền tín hiệu kỹ thuật số giữa 16 chân (8 chân đầu vào và 8 chân đầu ra). Đặc điểm của IC này là đầu ra tín hiệu luôn bằng hiệu điện thế cấp vào, trong khi hiệu điện thế của tín hiệu đầu vào có thể cao hơn hiệu điện thế cấp vào, do vậy ta có thể cấp nguồn điện 3.3v cho IC này và cho đầu vào có hiệu điện thế 5v từ mạch Arduino Nano và đầu ra sẽ có hiệu điện thế 3.3v phù hợp cho mạch ESP-12.



Các module cảm biến mà thiết bị sử dụng bao gồm:

- Cảm biến nhiệt độ và độ ẩm Aosong AM2301, với khả năng cảm biến độ ẩm từ 0-99% độ ẩm tương đối với độ chính xác 0.1% và khả năng cảm biến nhiệt độ trong khoảng từ -40 tới 80 độ C với độ chính xác 0.1 độ C.
- Cảm biến Sharp GP2Y1010AU0F cho việc cảm biến nồng độ bụi. Cảm biến có khả năng nhận dạng được nồng độ bụi nằm trong khoảng từ 0 tới 0.5ppm (một phần triệu)
- Cảm biến khí CO MQ-9 với khả năng phát hiện nồng độ khí CO trong khoảng từ 20 tới 2000ppm (một phần triệu)

Để hiển thị trạng thái hoạt động của thiết bị và số liệu đo được, thiết bị sử dụng một màn hình LCD 20x4 kết hợp với một mạch LCD I²C, giảm số lượng chân kết nối từ LCD tới mạch Arduino Nano từ 16 chân xuống còn 4 chân (hai chân nguồn và hai chân cho giao thức I²C)



Đèn báo bao gồm hai đèn LED ba màu (xanh, vàng, và đỏ). Một đèn hiển thị tình trạng hoạt động của thiết bị, đèn kia hiển thị chất lượng của không khí. Hai đèn này được mắc vào một IC 74HC595 trước khi kết nối với mạch Arduino Nano để giảm số chân cần thiết (6 chân xuống còn 3 chân). IC này là một thanh ghi 8 bit mà 8 bit này có thể được đọc ghi dưới dạng tuần tự (cần 2 chân và đọc/ghi theo thứ tự từng bit) hoặc dưới dạng song song (cần 8 chân, đọc/ghi từng bit bằng cách đọc/ghi từng chân). Lợi dụng điều này ta có thể điều khiển đèn LED mà chỉ cần phải kết nối ba chân (hai chân đọc/ghi tuần tự, 1 chân điều khiển bật/tắt)

Trang web

Phía máy chủ (backend)

Trang web được viết bằng ngôn ngữ Python. Có nhiều lý do để chọn Python thay vì các ngôn ngữ khác, một trong những điểm đó là việc Python có đặc tính dễ học, dễ viết và có cấu trúc trực quan hơn các ngôn ngữ khác.

Tuy nhiên chỉ có mỗi Python thì chưa đủ mà còn phải cần một nền tảng để hỗ trợ cho việc lập trình web. Chúng tôi quyết định lựa chọn nền tảng Flask vì Flask mặc dù đơn giản, tuy nhiên sự đơn giản đó cũng mang lại khả năng tùy biến để phù hợp cho ứng dụng cụ thể cao hơn so với các nền tảng khác (ví dụ như Django)

Về cơ sở dữ liệu, chúng tôi sử dụng hệ cơ sở dữ liệu MariaDB. MariaDB là một hệ cơ sở dữ liệu mới có mã nguồn gốc là hệ cơ sở dữ liệu MySQL. Ban đầu chúng tôi sử dụng hệ cơ sở dữ liệu SQLite, tuy nhiên sau đó thì chúng tôi quyết định chuyển sang MariaDB (vì nguyên tắc hoạt động của SQLite là lưu cơ sở dữ liệu lên một file, do vậy nó chỉ phù hợp với các dự án nhỏ). Thay vì sử dụng câu lệnh SQL trực tiếp, chúng tôi giao tiếp với hệ cơ sở dữ liệu thông qua thư viện Peewee. Peewee trừu tượng hóa giao tiếp giữa chương trình và hệ cơ sở dữ liệu, giúp việc giao tiếp trở nên dễ dàng, trực quan hơn và tránh được các lỗi như SQL injection.

Trong quá trình phát triển, chúng tôi nhận ra rằng có một số công việc như việc gửi email khá tốn thời gian. Vì vậy, thay vì để người dùng chờ thì chúng tôi sẽ đẩy việc đẩy vào một hàng đợi và sẽ có một tác vụ tách biệt với tác vụ phục vụ trang web thực hiện công việc trong hàng đợi ấy. Chương trình thực hiện tác vụ đó cũng được viết bằng Python và sử dụng Redis làm trình quản lý hàng đợi. Redis thực ra là một hệ cơ sở dữ liệu dạng khóa:giá trị, tuy nhiên Redis còn có nhiều tính năng khác, một trong số đó là việc hỗ trợ mô hình PubSub. Trong mô hình này, tiến trình phục vụ trang web sẽ gửi yêu cầu tác vụ vào nhiều kênh khác nhau, trong khi tiến trình còn lại sẽ lắng nghe yêu cầu nhận được ở các kênh đó. Ngay khi nhận được thông tin ở một kênh tiến trình sẽ tự động tạo luồng mới để xử lý công việc đó.

Đối với việc gửi tin nhắn và email, chúng tôi sử dụng hai dịch vụ của Amazon là Amazon SES (Simple Email Service) và Amazon SNS (Simple Notification Service). Hai dịch vụ này cho phép gửi email và tin nhắn thông qua việc gọi hàm API.

Ngoài những thư viện trên thì trang web còn sử dụng thêm các thư viện:

- bcrypt: mã hoá mật khẩu sử dụng thuật toán Bcrypt.
- boto3: API giúp chương trình giao tiếp với Amazon SES và Amazon SNS.
- phonenumberslite: thư viện hỗ trợ việc kiểm tra một số điện thoại có đúng với chuẩn hay không.

Phía người dùng (frontend)

Giao diện người dùng được viết bằng tổ hợp ba ngôn ngữ khác nhau: HTML, CSS và JavaScript.

Chúng tôi sử dụng Bootstrap làm nền tảng xây dựng bố cục trang web. Bootstrap là một thư viện CSS cung cấp sẵn các mẫu thiết kế thường dùng trong một trang web, giúp giảm bớt gánh nặng trong việc thiết kế bố cục mà vẫn đảm bảo một trang web đẹp và trực quan.

Nếu coi HTML và CSS là vẻ bề ngoài thì JavaScript là những thứ bên trong giúp một trang web hoạt động và phản ứng với các hành động của người dùng. JavaScript được sử dụng triệt để cho nhiều công việc khác nhau như:

- Nạp, cập nhật dữ liệu và thay đổi dữ liệu trong trang web mà không cần phải khởi tạo lại trang (sử dụng JQuery)
- Hiện thị bản đồ và heatmap (sử dụng Google Maps JavaScript API)
- Vẽ biểu đồ sử dụng thư viện Highstocks.
- Cập nhật dữ liệu theo thời gian thực sử dụng thư viện Sockets.IO.
- Một số thư viện khác:
 - Gợi ý và nhập nhân sử dụng bootstrap-tagsinput và typeahead.js
 - Hiện thị thời gian sử dụng moment.js
 - Chỉnh sửa ảnh sử dụng cropit.js (để người dùng có thể đặt avatar)
 - Lựa chọn địa điểm khi chỉnh sửa thông tin trạm sử dụng jquery-locationpicker-plugin

SỐ LIỆU/ KẾT QUẢ NGHIÊN CỨU

Sau một thời gian nghiên cứu và chế tạo, dự án đã hoàn thiện được bộ sản phẩm đáp ứng được nhu cầu đề ra ban đầu với các chức năng sau:

Thiết bị quan trắc

- Tự động nhận biết các chỉ số của không khí bao gồm nhiệt độ, độ ẩm không khí, nồng độ bụi và nồng độ khí CO trong không khí thông qua các cảm biến.
- Hiển thị các thông số đo được trên một màn hình LCD kích thước 20x4.
- Cảnh báo với người dùng thông qua các kênh:
 - Đèn LED ba màu (xanh, vàng, đỏ ứng với mức độ cảnh báo)
 - Email và SMS thông qua trang web
- Kết nối wifi thông qua module ESP-12 và tự động gửi thông tin lên máy chủ định kì trong một khoảng thời gian nhất định là 5 phút một lần.
- Cho phép người dùng chỉnh sửa các thông số của thiết bị ví dụ như tên và mật khẩu của trạm phát wifi mà thiết bị kết nối tới, địa chỉ IP tĩnh, vân vân.

Trang web

- Cho phép hiển thị thông tin của các trạm một cách trực quan dưới dạng bản đồ (dựa trên Google Maps) và biểu mẫu, người dùng có thể chọn thời gian mà biểu mẫu biểu diễn. Sử dụng thư viện Socket.IO, thông tin cũng sẽ được cập nhật theo thời gian thực.
- Hiển thị các trạm dưới dạng heatmap (bản đồ nhiệt, trên bản đồ sẽ có các vùng màu xanh hoặc vàng hoặc đỏ biểu thị mức độ ô nhiễm ở vùng đó).
- Cho phép người dùng tạo tài khoản để phục vụ nhiều mục đích khác nhau, như:
 - Cho phép người dùng thêm mới, quản lý và sửa đổi thông tin của các trạm quan trắc thuộc quyền sở hữu của mình.
 - Cho phép người dùng đăng kí nhận tin khi một trạm quan trắc nào đó có số đo ở mức nguy hiểm thông qua việc gửi email và nhắn tin SMS tới người dùng.

- Cho phép ẩn một trạm, nghĩa là ngoài người chủ ra thì không ai có thể truy cập và xem thông tin về trạm đó.
- Cung cấp một số hàm API cho việc tích hợp trang web với các ứng dụng ngoài
- Cho phép truy cập vào cơ sở dữ liệu thông qua ba cách:
 - Thông qua các hàm API đối với các truy vấn nhỏ.
 - Cho phép xuất và tải về số liệu của các trạm dưới các định dạng JSON và CSV
 - Cho phép người dùng thực thi truy vấn trực tiếp trong cơ sở dữ liệu của trang web thông qua việc viết các truy vấn SQL trả về kết quả.
- Cho phép phân loại các trạm theo các nhãn (tag)
- Cho phép tìm các trạm theo tên và theo nhãn, tìm nhãn và người dùng theo tên
- Hiện tại thì trang web đã có thể truy cập tại địa chỉ <https://e3.tuankiet65.moe>

PHÂN TÍCH SỐ LIỆU/ KẾT QUẢ VÀ THẢO LUẬN

Điểm yếu

Nghiên cứu đã hoàn thiện trong việc thiết kế mô hình hệ thống theo dõi và cảnh báo tự động về chất lượng không khí. Tuy nhiên, nghiên cứu vẫn còn tồn đọng những điểm chưa hoàn thiện:

Khoảng cách giao tiếp

Mô hình hiện tại sử dụng sóng Wifi để thực hiện việc truyền tin không dây. Tuy nhiên, khi sử dụng các thiết bị thu phát quy mô nhỏ, khoảng cách lớn nhất giữa hai thiết bị chỉ đạt từ 46m (trong nhà) cho tới 92m (ngoài trời) [Bradley, 2016]. Sử dụng các thiết bị thu phát quy mô lớn hơn thì khoảng cách có thể đạt tới hàng chục km, tuy nhiên kèm theo đó là chi phí đầu tư cho những trạm này rất lớn.

Trong quá trình nghiên cứu mở rộng, tôi đã thực hiện tìm hiểu về các giải pháp xử lý vấn đề này, và đã tìm ra một số giải pháp khác nhau. Điểm chung giữa những giải pháp này là chúng chỉ có thể thực hiện truyền tin giữa hai thiết bị mà không phải là truyền trực tiếp từ thiết bị lên mạng Internet giống như Wifi, vì vậy mô hình hoạt động sẽ bao gồm nhiều máy con và một máy chủ thực hiện nhận tin và truyền lại qua mạng Internet.

- Semtech LoRa: LoRa là công nghệ LPWAN (Low Power Wide Area Network) cho phép truyền tin sử dụng sóng RF ở băng tần 868MHz [Libelium Comunicaciones Distribuidas 2016a]. Lợi thế của việc truyền tin ở băng tần này là khoảng cách truyền tin xa hơn hẳn Wifi (ở băng tần 2.4GHz), vào khoảng từ 1km (có nhà chắn) cho tới 21km (không nhà chắn) và có mức sử dụng năng lượng thấp [Libelium Comunicaciones Distribuidas 2016b]. Bù lại khoảng cách xa là tốc độ truyền tin thấp, tuy nhiên việc này có thể được khắc phục.
- Nordic nRF24L01: nRF24L01 cũng sử dụng sóng vô tuyến ở băng tần 2.4GHz. Module này có giá thành rẻ (dưới 100-200 ngàn) và khá phổ biến

ở Việt Nam, có tốc độ truyền từ 250kbit/s trở lên, tuy nhiên khoảng cách của module này chỉ từ 1km trở xuống (trên lý thuyết)

Việc thiếu các cảm biến

KẾT LUẬN

chúng tôi đã chế tạo bla bla bla,

TÀI LIỆU THAM KHẢO

References

Balmes, J. R.; Fine, J. M. and Sheppard, D. (1987). *Symptomatic Bronchoconstriction after Short-Term Inhalation of Sulfur Dioxide*, American Review of Respiratory Disease 136 : 1117-1121.

Bernstein, J. A.; Alexis, N.; Barnes, C.; Bernstein, I. L.; Nel, A.; Peden, D.; Diaz-Sanchez, D.; Tarlo, S. M.; Williams, P. B. and Bernstein, J. A. (2004). *Health effects of air pollution*, Journal of Allergy and Clinical Immunology 114 : 1116-1123.

Kampa, M. and Castanas, E. (2008). *Human health effects of air pollution*, Environmental Pollution 151 : 362-367.

Libelium Comunicaciones Distribuidas, S.L. **(2016b)**. *Extreme Range Links: LoRa 868 / 900MHz SX1272 LoRa module for Arduino Waspote and Raspberry Pi*, .

Libelium Comunicaciones Distribuidas, S.L. **(2016a)**. *Waspote-LoRa-868MHz_915MHz-SX1272 Networking Guide*, .