Báo cáo môn Project III

Đề tài: Thiết bị giám sát y tế.

Giáo viên hướng dẫn: TS Nguyễn Hữu Đức

Sinh viên thực hiện: Nguyễn Công Trưởng MSSV:20173431

# Đặt vấn đề, giới thiệu đề tài.

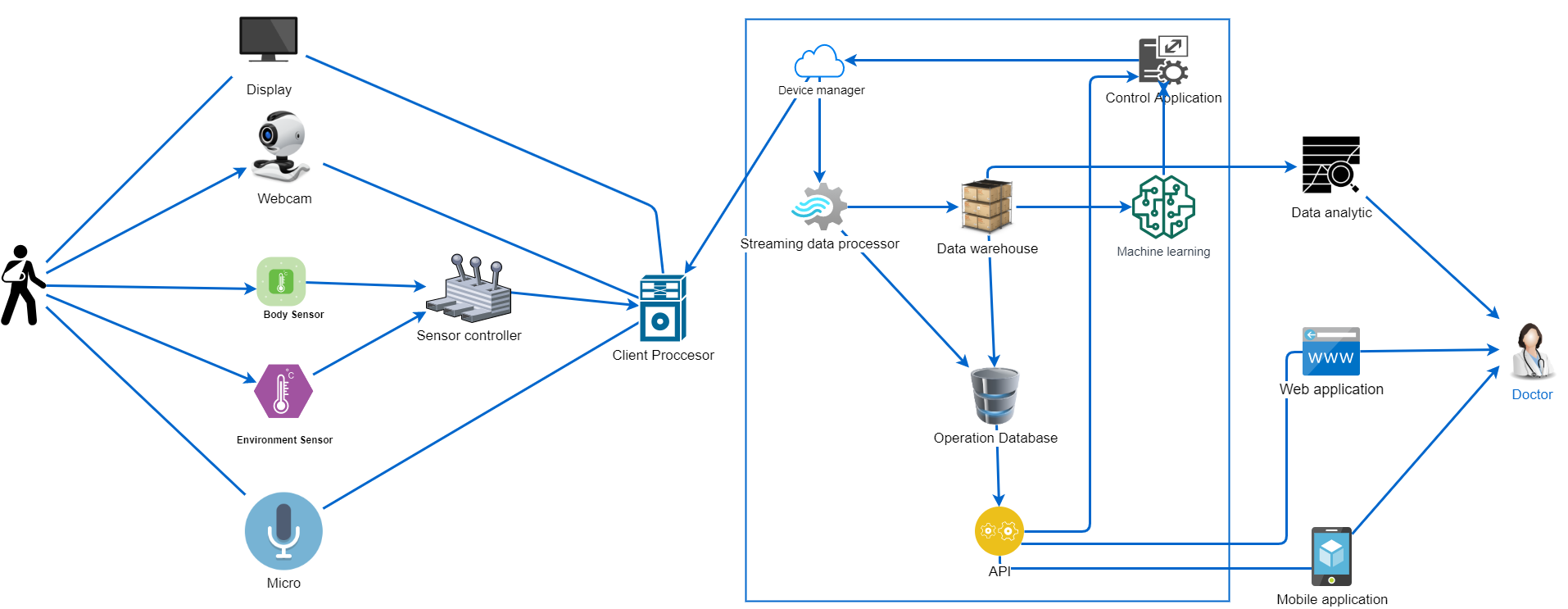
Hiện nay, hệ thống giám sát chăm sóc sức khỏe di động với các công nghệ mới đang trở thành mối quan tâm lớn của nhiều quốc gia trên toàn thế giới.

Sự ra đời của công nghệ Internet of Things (IoT) tạo điều kiện thuận lợi cho sự phát triển của chăm sóc sức khỏe, tư vấn trực tiếp đến y tế từ xa nhất là trong điều kiện tình hình dịch bệnh COVID như hiện nay.

Một thiết bị có thể giúp bác sĩ có các chỉ số cần thiết để đưa ra các chẩn đoán từ xa cho bệnh nhân đang cách li, hoặc không thể đi lại, từ đó giảm thiểu sự quá tải của bệnh viện.

Trong báo cáo này nhóm 5 sẽ đề xuất một hệ thống giám sát sức khỏe người bệnh với các chỉ số cơ bản để người có chuyên môn có thể đưa ra các tư vấn ý tế cần thiết

# Thiết kế hệ thống.



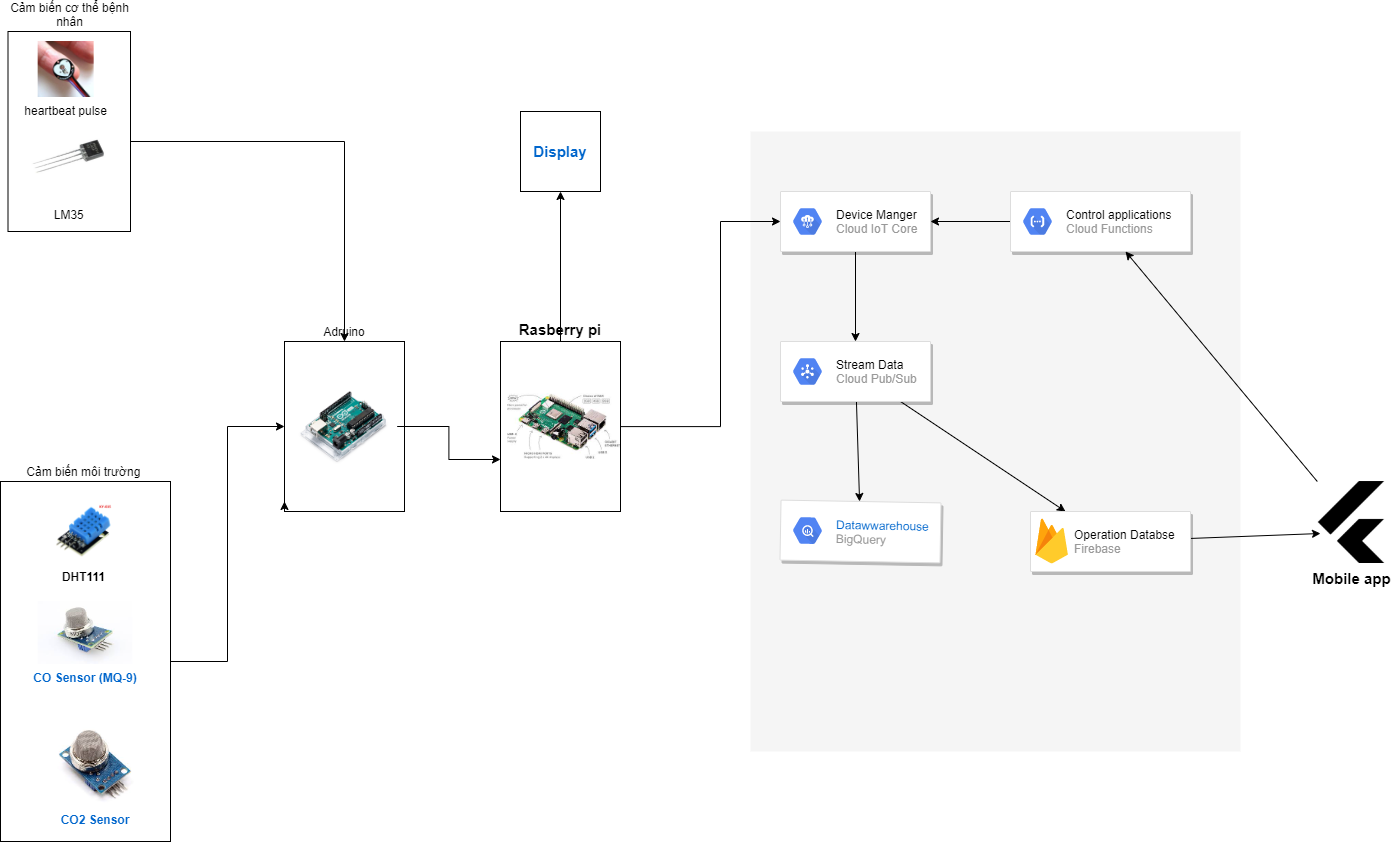
Hệ thống được thiết kế chia làm 3 tầng:

* Tầng khởi đầu (phía bệnh nhân): Bao gồm các cảm biến, trình điều khiển và thiết bị xử lý tính toán.
* Tầng trung gian: Nơi lưu trữ dữ liệu, cung cấp các API tính toán.
* Tầng đích (phía bác sĩ): Cung cấp giao diện ứng dụng cho bác sĩ để đưa ra các nhận định, hỗ trợ bệnh nhân.

|  |  |
| --- | --- |
| Tên thành phần | Giải thích |
| Body Sensor | Các cảm biến đo các chỉ số sinh lí của người bệnh: nhịp tim, nhiệt độ cơ thể.., |
| Environment Sensor | Các cảm biến đo điều kiện nơi chăm sóc bệnh nhân: nhiệt độ phòng, áp suất, lượng khí CO, lượng khí CO2... |
| Webcam | Giúp bệnh nhân giao tiếp với bác sĩ |
| Micro |
| Màn hình |
| Client Processor | Bộ xử lý phía client |
| Device manager | Trình quản lí thiết bị |
| Streaming Data Processor | Quản lý hướng đến của dữ liệu |
| Data warehouse | Nơi lưu trữ dữ liệu lâu dài phục vụ analytic, query |
| Operation Database | Database sử dụng cho ứng dụng |
| Control application | Các chương trình dùng để điều khiển thiết bị |
| Machine Learning | Các chương trình sử dụng học máy để đưa ra các cảnh báo sơ bộ |
| API | Các API hỗ trợ giao diện cho người sử dụng |
| Web Application | Đưa ra các báo cáo về thông số cho bác sĩ |
| Mobile Application |
| Data analytics |

# Triển khai.

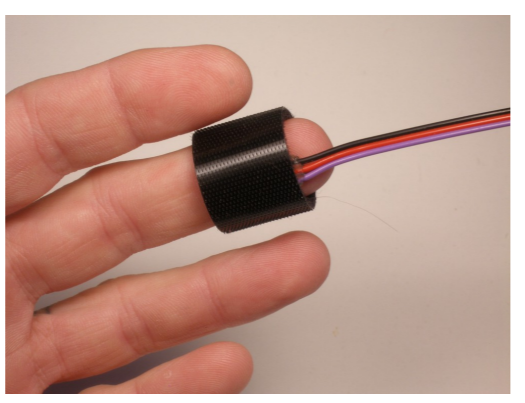
## **Sơ đồ thiết kế triển khai.**



## **Phân tích chi tiết các thành phần hệ thống:**

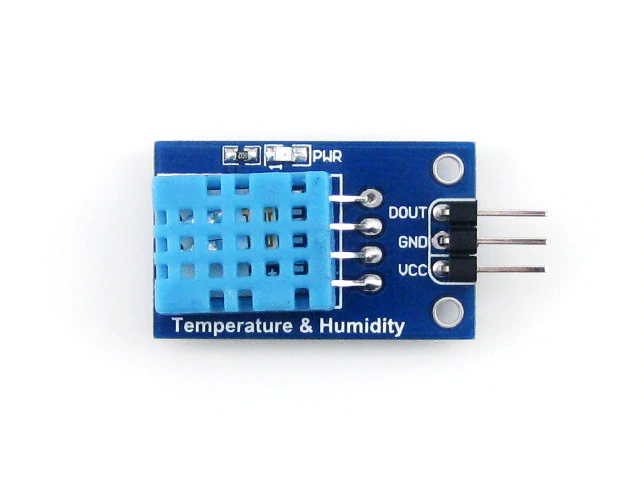
Các cảm biến

1. **Heartbeat pulse**: Cảm biến nhịp tim là một cảm biến nhịp tim plug-and-play được thiết kế tốt cho Arduino. Cảm biến kẹp vào đầu ngón tay hoặc dái tai và cắm ngay vào Arduino bằng một số cáp. Nó cũng bao gồm một thư viện giám sát mã nguồn mở lập biểu đồ xung của bạn trong thời gian thực.



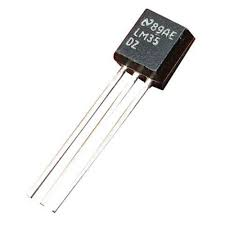
*Hình 1: Cảm biến Heartbeat pulse*

1. **DHT11:** Cảm biến độ ẩm và nhiệt độ DHT11 Temperature Humidity Sensor là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1 wire (giao tiếp digital 1 dây truyền dữ liệu duy nhất). Bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp bạn có được dữ liệu chính xác mà không phải qua bất kỳ tính toán nào.



*Hình 2: Cảm biến DHT11*

1. **LM35:** Cảm biến nhiệt độ nhỏ gọn, độ chính xác cao hơn DHT11, chúng ta chọn cảm biến này để đo nhiệt độ cơ thể



*Hình 3: Cảm biến LM35*

1. **MQ-9:** thích hợp để phát hiện LPG, CO và CH4. Nhờ độ nhạy cao và thời gian phản hồi nhanh, các phép đo có thể được thực hiện nhanh chóng**.**

****

*Hình 4: Cảm biến MQ-9*

1. **MQ-135:** Đối với các hệ thống kiểm soát chất lượng không khí, các cảm biến khí MQ-135 được sử dụng để phát hiện NH3, Nicotine, Benzen, Khói và CO2 cũng như đo lường. Mô-đun cảm biến MQ-135 đi kèm với chân kỹ thuật số cho phép cảm biến này hoạt động ngay cả khi không có bộ vi điều khiển và có lợi cho việc phát hiện các loại khí cụ thể. Khí trong PPM được tính bằng các chân analog. Chân tương tự được cung cấp bởi TTL và hoạt động trên 5 V, do đó nó có thể được sử dụng với hầu hết các bộ vi điều khiển hiện đại.

` 

*Hình 5: Cảm biến MQ135*

Trong project này chúng ta sử dụng Arduino Uno để quản lý các cảm biến,

Raspberry Pi 3 để hỗ trợ các kết nối liên quan đến IOT với hệ thống Google Cloud Service.

## **Các công nghệ sử dụng:**

1. Sử dụng python làm ngôn ngữ giao tiếp với server.
2. Hệ thống sử dụng quản lý thiết bị IOT bằng hệ thống Google Cloud Service:

* Cloud IOT Core: Đóng vai trò là Device Manager, dùng để quản lý trạng thái các thiết bị trên hệ thống.
* Cloud Pub/Sub: Hoạt động theo cơ chế publisher/subscriber, điều phối đường đi của data streaming
* Google BigQuery: Dùng để lưu trữ dữ liệu lớn, sử dụng như một Data warehouse
* Firebase: Cơ sở dữ liệu thời gian thực, là một operation database
* Cloud Function: Chạy các service hỗ trợ điều khiển thiết bị, tạo các API cần thiết.

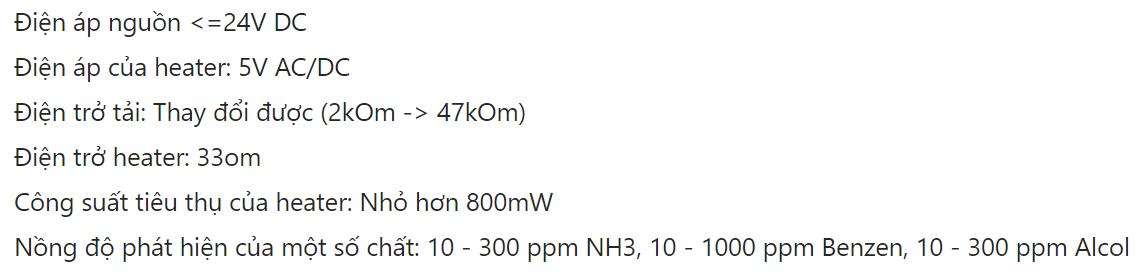
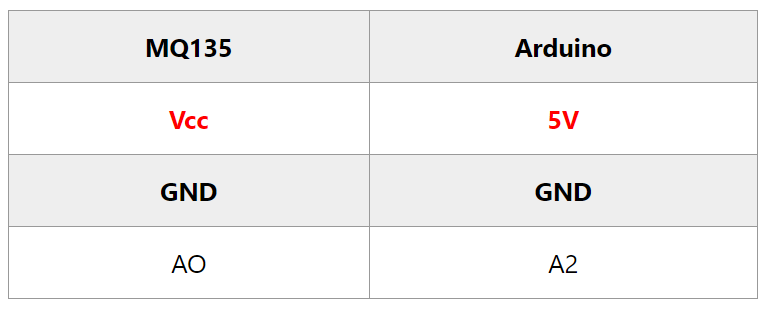
1. Mobile App được viết bằng Flutter.

## **Kịch bản giao tiếp:**

## 

Hình 6: Sơ đồ kịch bản giao tiếp

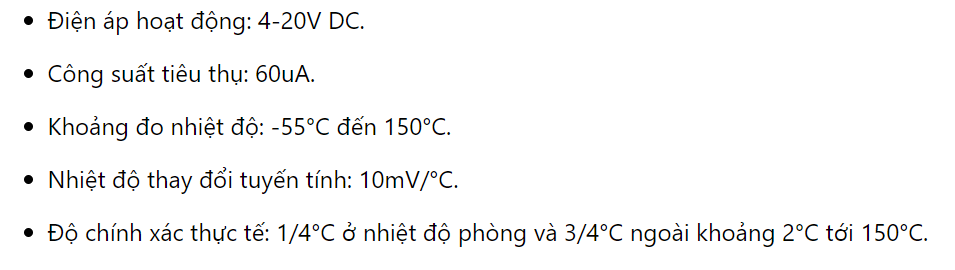
**Cảm biến MQ135:**

* + **Giới thiệu**: Cảm biến này có thể nhận biết được các chất khí như NH3, Nox, Ancol, ``Benzen, Khói, gas, CO2......
  + **Thông số kỹ thuật:**
  + 
  + **Sơ đồ ghép nối:**
  + 

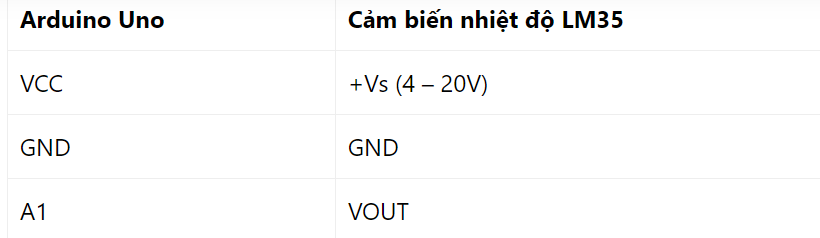
## Code triển khai: Sử dụng thư viện MQ135 của tác giả GeorgK

**Cảm biến LM35:**

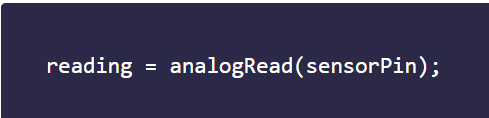
* **Giới thiệu:** Cảm biến nhiệt độ LM35 là bộ cảm biến nhiệt mạch tích hợp chính xác cao mà điện áp đầu ra của nó tỷ lệ tuyến tính với nhiệt độ theo thang độ Celsius.
* **Thông số kỹ thuật:**



* **Sơ đồ ghép nối**

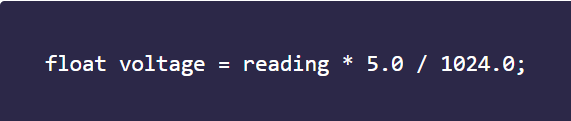
****

* **Triển khai**



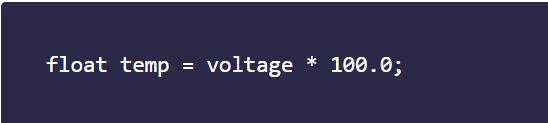
Hàm analogRead() có nhiệm vụ đọc giá trị điện áp từ một chân Analog (ADC), hàm này luôn trả về 1 số nguyên nằm trong khoảng từ 0 đến 1023 tương ứng với thang điện áp (mặc định) từ 0 đến 5V. Hàm analogRead() cần 100 micro giây để thực hiện.

Hàm analogRead() có nhiệm vụ đọc giá trị điện áp từ một chân Analog (ADC), hàm này luôn trả về 1 số nguyên nằm trong khoảng từ 0 đến 1023 tương ứng với thang điện áp (mặc định) từ 0 đến 5V. Hàm analogRead() cần 100 micro giây để thực hiện.

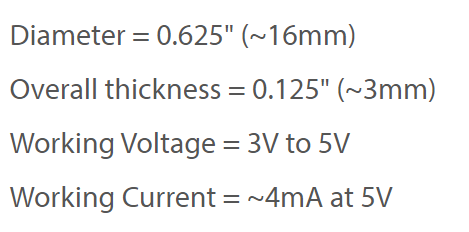


Công thức tính ra giá trị hiệu điện thế từ giá trị cảm biến (đơn vị Volt) Voltage = giá trị điện áp từ cảm biến chia cho mức analog cao nhất (1024) rồi nhân với mức điện áp 5V.

Như ở trên ta thấy nhiệt độ thay đổi tuyến tính 10mV/°C nên đổi từ Vol sang °C thì ta chỉ cần nhân giá trị điện thế với 100 là ra nhiệt độ.



**Cảm biến heartbeat pulse:**

* **Giới thiệu:** Các biến nhịp tim dựa trên nguyên tắc photoplethysmography(PPG). Sự thay đổi của khối lượng máu qua bất kỳ cơ quan nào của cơ thể sẽ tạo ra một sự thay đổi về cường độ ánh sáng thông qua cơ quan đó (vùng có mạch máu). Mỗi lần tim đập, dòng chảy của lượng máu sẽ thay đổi và từ đó, ánh sáng lượng hồng ngoại hấp thụ bởi ánh sáng sẽ thay đổi, tạo ra xung thay đổi trên thiết bị thu hồng ngoại. Cảm biến này được dùng nhiều trong thiết bị theo dõi nhịp tim giá rẻ.
* **Thông số kĩ thuật:**
* ****

## **Đánh giá bảo mật**:

Bảo mật là mối quan tâm hàng đầu khi triển khai và quản lý các thiết bị IoT. Hệ thống sử dụng các cơ chế sau đây để bảo mật:

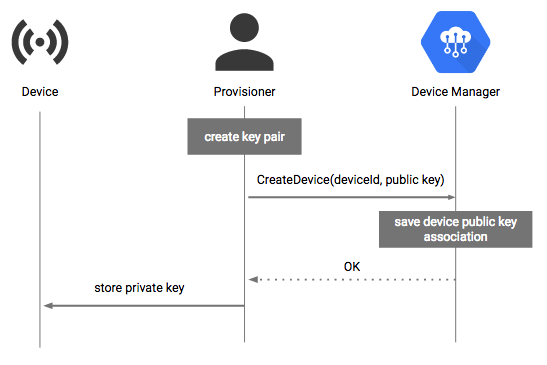
1. Mỗi thiết bị được sử dụng cặp khóa public key/private key để xác thực sử dụng JSON Web Token (JWTs, [RFC 7519](https://tools.ietf.org/html/rfc7519))

- Hạn chế vùng tác động của các cuộc tấn công, nếu một cặp khóa bị lộ chỉ ảnh hưởng đến một thiết bị chứ không phải toàn bộ hệ thống.

- JWT có hiệu lực trong một khoảng thời gian nhất định, vì vậy bất kỳ khóa nào bị đánh cắp sẽ hết hạn.

1. Sử dụng kết nối TLS 1.2 trong kết nối, sử dụng root certificate authorities

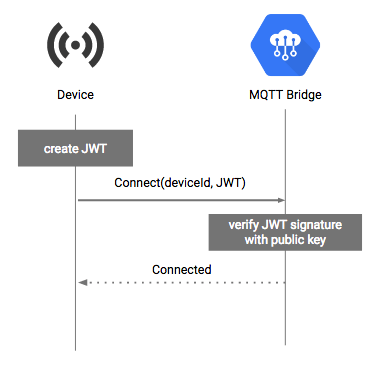
**Tạo khóa:**



1.Cặp khóa công khai-riêng tư được tạo bởi người cung cấp.

1. Người cung cấp tạo thiết bị, chỉ định khóa công khai vừa tạo. Điều này sẽ được sử dụng để xác minh danh tính của thiết bị.
2. Trình quản lý thiết bị lưu trữ tài nguyên thiết bị và khóa công khai.
3. Trình quản lý thiết bị thông báo thiết bị được khởi tạo
4. Khóa cá nhân được lưu trữ trên thiết bị để sử dụng sau này để xác thực

**Xác thực:**

****

1. Thiết bị chuẩn bị JSON Web Token được kí bằng private key.
2. Khi kết nối với MQTT bridge, thiết bị hiển thị JWT dưới dạng mật khẩu trong tin nhắn MQTT CONNECT. Nội dung tên người dùng bị bỏ qua.
3. MQTT Bridge xác thực JWT bằng public key
4. MQTT Bridge chấp nhận kết nối
5. Kết nối bị đóng khi JWT hết hạn

**Các chuẩn bảo mật:**

**JWT RS256**

**JWT ES256**