**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**CHUYÊN NGÀNH: AN TOÀN THÔNG TIN**

**ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG INTERNET OF THINGS**

**HỖ TRỢ QUAN TRẮC MÔI TRƯỜNG**

Giảng viên hướng dẫn: **Th.S PHAN CHÍ TÙNG**

Sinh viên thực hiện: **PHẠM HÒA MÂU**

Số thẻ sinh viên: **102160054**

Lớp**: 16T1**

**Đà Nẵng, 12/2020**

|  |  |
| --- | --- |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc |

**NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP**

**1. Thông tin chung:**

1. Họ và tên sinh viên: Phạm Hòa Mâu

2. Lớp: 16T1 Số thẻ SV: 102160054

3. Tên đề tài: Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường.

4. Người hướng dẫn: Th.S Phan Chí Tùng

**II. Nhận xét, đánh giá đồ án tốt nghiệp:**

1. Về tính cấp thiết, tính mới, khả năng ứng dụng của đề tài: (điểm tối đa là 2đ)

………………………………………………………………………………………..…………………………………………………………………………..

2. Về kết quả giải quyết các nội dung nhiệm vụ yêu cầu của đồ án: (điểm tối đa là 4đ)

………………………………………………………………………………………..…………………………………………………………………………..

3. Về hình thức, cấu trúc, bố cục của đồ án tốt nghiệp: (điểm tối đa là 2đ)

………………………………………………………………………………………..…………………………………………………………………………..

4. Đề tài có giá trị khoa học/ có bài báo/ giải quyết vấn đề đặt ra của doanh nghiệp hoặc nhà trường: (điểm tối đa là 1đ)

…………………………………………………………………………………

…………………………………………………………………………………

5. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:

…………………………………………………………………………………

…………………………………………………………………………………

**III. Tinh thần, thái độ làm việc của sinh viên:** (điểm tối đa 1đ)

…………………………………………………………………………………

**IV. Đánh giá:**

1. Điểm đánh giá: ……../10 (lấy đến 1 số lẻ thập phân)

2. Đề nghị:  Được bảo vệ đồ án  Bổ sung để bảo vệ  Không được bảo vệ

|  |  |
| --- | --- |
|  | *Đà Nẵng, ngày tháng năm 2020* |
|  | **Người hướng dẫn** |

**NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN**

...........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

Đà Nẵng, ngày … tháng … năm 2020

Giảng viên hướng dẫn

**Th.S Phan Chí Tùng**

NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN

..........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

Đà Nẵng, ngày … tháng … năm 2020

Giảng viên phản biện

**PHIẾU DUYỆT ĐỒ ÁN TỐT NGHIỆP**

**I. Phần dành cho Sinh viên**

1. Họ và tên:Phạm Hòa Mâu 2. Mã Sinh viên: 102160054 3. Lớp: 16T1

4. Tên đề tài: Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường.

5. Số điện thoại: 0398636611 6. E-mail: mauphamhoa@gmail.com

7. Họ và tên GVHD: Th.S Phan Chí Tùng

**II. Phần dành cho Hội đồng**

|  |  |  |
| --- | --- | --- |
| **STT** | **Nội dung đánh giá** | **Kết luận** |
|  | Trình bày báo cáo theo đúng mẫu qui định của Khoa |  |
|  | Không có sự sao chép nội dung báo cáo và chương trình đã có |  |
|  | Biên dịch mã nguồn và chạy được chương trình |  |
|  | Có kịch bản thực hiện với dữ liệu thử nghiệm |  |
|  | Kết quả thực hiện chương trình đúng theo báo cáo |  |
|  | Có sự đóng góp, phát triển của tác giả trong đồ án |  |

**Ý kiến khác:**

........................................................................................................................................................................................................................................................................................................................................................................................

**Kết luận:** 🞎 Đạt yêu cầu 🞎 Phải sửa chữa lại 🞎 Không đạt yêu cầu

Đà Nẵng, ngày tháng 12 năm 2020

|  |  |
| --- | --- |
| **Chủ tịch Hội đồng**  (Ký và ghi họ tên) | **Cán bộ duyệt kiểm tra**  (Ký và ghi họ tên) |
|  |  |

# TÓM TẮT

Tên đề tài: Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường.

Sinh viên thực hiện: Phạm Hòa Mâu

Số thẻ SV: 102160054 Lớp: 16T1

Hệ thống này sinh ra nhằm giải quyết 3 vấn đề chính:

* Quan trắc môi trường tự động và liên tục: Hệ thống bao gồm các trạm quan trắc được tích hợp các cảm biến giúp đo đạc các chỉ số liên quan đến môi trường một cách tự động và dữ liệu được gửi liên tục theo thời gian thực về hệ thống thông qua hạ tầng mạng di động.
* Đảm bảo tính minh bạch của dữ liệu quan trắc: Hệ thống kết hợp công nghệ Blockchain với IoT, nhờ tính năng bất biến dữ liệu trong Blockchain giúp đảm bảo tính toàn vẹn và minh bạch của dữ liệu quan trắc trước sự thao túng của các tác nhân bên ngoài như nhóm lợi ích hay sự chi phối của đồng tiền.
* Giám sát và truy xuất dữ liệu: Hệ thống cho phép người dùng giám sát dữ liệu quan trắc thông qua các biểu đồ được cập nhật theo thời gian thực của các trạm quan trắc. Ngoài ra, hệ thống còn bao gồm một ứng ụng di động cho phép người dùng có thể truy xuất dữ liệu của các trạm theo thời gian thực và các khoảng thời gian trong quá khứ.

|  |  |
| --- | --- |
| ĐẠI HỌC ĐÀ NẴNG  **TRƯỜNG ĐẠI HỌC BÁCH KHOA**  KHOA CÔNG NGHỆ THÔNG TIN | **CỘNG HÒA XÃ HÔI CHỦ NGHĨA VIỆT NAM**  Độc lập - Tự do - Hạnh phúc |

# NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Phạm Hòa Mâu Số thẻ sinh viên: 102160054

Lớp: 16T1 Khoa: Công nghệ thông tin Ngành: An toàn thông tin

1. Tên đề tài đồ án:

Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường.

1. Đề tài thuộc diện:  Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện.
2. Các số liệu và dữ liệu ban đầu: Không có.
3. Nội dung các phần thuyết minh:

**Giới thiệu đề tài:** Phần mở đầu của luận văn, giới thiệu về lý do thực hiện đề tài, đồng thời giới thiệu sơ lược về đề tài và mục tiêu phải đạt được.

**Phần 1**. Giới thiệu thực trạng ô nhiễm môi trường và đề xuất giải pháp

**Phần 2**. Tổng quan về các linh kiện, cảm biến thường được dùng trong các trạm quan trắc. Giới thiệu về kiến trúc microservice và những ưu điểm của kiến trúc microservice so với kiến trúc nguyên khối. Giới thiệu về công nghệ Blockchain.

**Phần 3**. Triển khai thực nghiệm, đánh giá các kết quả đạt được, kết luận và hướng phát triển.

1. Các bản vẽ, đồ thị ( ghi rõ các loại và kích thước bản vẽ ): Không có.
2. Họ tên người hướng dẫn: Th.S Phan Chí Tùng
3. Ngày giao nhiệm vụ đồ án:  30/08/ 2020
4. Ngày hoàn thành đồ án: 30/11/ 2020

|  |  |
| --- | --- |
|  | Đà Nẵng, ngày tháng 12 năm 2020 |
|  |  |
| **Trưởng Bộ môn** …………………….. | **Người hướng dẫn** |

# LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành và sâu sắc đến các thầy cô trong Khoa Công nghệ thông tin, cũng như tất cả các thầy cô trong trường Đại học Bách khoa– Đại học Đà Nẵng đã dìu dắt, dạy dỗ và truyền đạt kiến thức, kinh nghiệm quý báu của mình trong suốt quá trình em học tập và nghiên cứu tại trường.

Em xin bày tỏ tình cảm và lòng biết ơn chân thành tới Th.S Phan Chí Tùng, người đã từng bước hướng dẫn, giúp đỡ tận tình và tạo điều kiện thuận lợi nhất cho em trong quá trình thực hiện đồ án tốt nghiệp của mình. Nhờ đó em đã hoàn thành đồ án đúng tiến độ và tích lũy cho mình nhiều kiến thức quý báu.

Trong quá trình thực hiện chắc hẳn sẽ không tránh khỏi những thiếu sót nhất định. Em rất mong nhận được sự cảm thông, góp ý và tận tình chỉ bảo của quý thầy cô để đề tài của em được hoàn thiện hơn.

Một lần nữa em xin chân thành cảm ơn!

**Phạm Hòa Mâu**

# CAM ĐOAN

Em xin cam đoan:

1. Nội dung trong đồ án này là do em thực hiện dưới sự hướng dẫn trực tiếp của Th.S Phan Chí Tùng.
2. Các tham khảo dùng trong đồ án đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố.

Nếu có những sao chép không hợp lệ, vi phạm quy chế đào tạo, em xin chịu hoàn toàn trách nhiệm.

Đà Nẵng, ngày tháng 12 năm 2020

Sinh viên thực hiện

**Phạm Hòa Mâu**

MỤC LỤC

[TÓM TẮT](#_Toc58189529)

[NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP](#_Toc58189530)

[LỜI CẢM ƠN i](#_Toc58189531)

[CAM ĐOAN ii](#_Toc58189532)

[DANH SÁCH CÁC BẢNG vi](#_Toc58189533)

[DANH SÁCH CÁC HÌNH VẼ vii](#_Toc58189534)

[DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT ix](#_Toc58189535)

[GIỚI THIỆU ĐỀ TÀI 1](#_Toc58189536)

[**1.** **Lý do chọn đề tài** 1](#_Toc58189537)

[**2.** **Mục đích của đề tài** 1](#_Toc58189538)

[**3.** **Nội dung của đề tài** 1](#_Toc58189539)

[**4.** **Bố cục của đề tài** 2](#_Toc58189540)

[CHƯƠNG 1: TỔNG QUAN VỀ TRẠM QUAN TRẮC MÔI TRƯỜNG 3](#_Toc58189541)

[**1.** **Phát triển phần cứng** 3](#_Toc58189542)

[**1.1.** **Nguyên lý hoạt động** 3](#_Toc58189543)

[**1.2.** **Sơ đồ khối** 3](#_Toc58189544)

[**1.3.** **Linh kiện** 3](#_Toc58189545)

[**2.** **Phát triển firmware** 13](#_Toc58189546)

[**2.1.** **Sơ đồ thuật toán** 13](#_Toc58189547)

[**2.2.** **Phầm mềm hỗ trợ** 13](#_Toc58189548)

[CHƯƠNG 2: GIỚI THIỆU VỀ KIẾN TRÚC MICROSERVICE 14](#_Toc58189549)

[**1.** **Kiến trúc microservice là gì?** 14](#_Toc58189550)

[**2.** **Ưu điểm và nhược điểm của microservice** 15](#_Toc58189551)

[**3.** **Microservice và Docker** 16](#_Toc58189552)

[CHƯƠNG 3: TÌM HIỂU CÔNG NGHỆ BLOCKCHAIN 17](#_Toc58189553)

[**1.** **Tổng quan về Blockchain** 17](#_Toc58189554)

[**1.1.** **Hàm băm** 18](#_Toc58189555)

[**1.2.** **Khối (Block) và dữ liệu khối** 18](#_Toc58189556)

[**1.3.** **Blockchain ứng dụng trong giao dịch** 19](#_Toc58189557)

[**1.4.** **Nút của Blockchain** 19](#_Toc58189558)

[**2.** **Nguyên lý hoạt động của Blockchain** 19](#_Toc58189559)

[**2.1.** **Các phiên bản của Blockchain** 20](#_Toc58189560)

[**2.2.** **Cơ bản về Bitcoin** 20](#_Toc58189561)

[**2.3.** **Cơ chế đào coin và hiểu sâu hơn về Blockchain** 21](#_Toc58189562)

[**3.** **Triển khai hệ thống Blockchain hỗ trợ quan trắc môi trường** 22](#_Toc58189563)

[**3.1.** **Giới thiệu về Hyperledger Fabric** 22](#_Toc58189564)

[**3.2.** **Kiến trúc đơn giản của một mạng Hyperledger Fabric** 22](#_Toc58189565)

[**4.** **Ưu điểm của hệ thống Blockchain** 30](#_Toc58189566)

[**4.1.** **Tiết kiệm chi phí** 30](#_Toc58189567)

[**4.2.** **Tính bảo mật cao** 30](#_Toc58189568)

[**4.3.** **Dữ liệu không bị mất** 30](#_Toc58189569)

[**4.4.** **Giao dịch xuyên biên giới** 30](#_Toc58189570)

[**4.5.** **Đảm bảo sự minh bạch** 30](#_Toc58189571)

[**4.6.** **Giảm thiểu sự chậm trễ khi xử lý giao dịch** 30](#_Toc58189572)

[CHƯƠNG 4: TRIỀN KHAI THỰC NGHIỆM 32](#_Toc58189573)

[**1.** **Mô tả hệ thống** 32](#_Toc58189574)

[**2.** **Trạm quan trắc môi trường** 32](#_Toc58189575)

[**3.** **Kiến trúc microservice** 32](#_Toc58189576)

[**4.** **Hệ thống Blockchain** 33](#_Toc58189577)

[**5.** **Ứng dụng web và di động** 34](#_Toc58189578)

[CHƯƠNG 5: KẾT QUẢ VÀ ĐÁNH GIÁ 36](#_Toc58189579)

[**1.** **Đánh giá độ chính xác của phần cứng và phần mềm** 36](#_Toc58189580)

[**1.1.** **Phần cứng** 36](#_Toc58189581)

[**1.2.** **Phần mềm** 38](#_Toc58189582)

[**2.** **Ngăn chặn tấn công từ các trạm quan trắc giả mạo** 39](#_Toc58189583)

[**3.** **Ngăn chặn phá hủy dữ liệu trên time-series database** 39](#_Toc58189584)

[CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 41](#_Toc58189585)

[**1.** **Kết luận** 41](#_Toc58189586)

[**2.** **Hướng phát triển** 41](#_Toc58189587)

[TÀI LIỆU THAM KHẢO 42](#_Toc58189588)

# DANH SÁCH CÁC BẢNG

[Bảng 1: Sơ đồ nối chân cảm biến DHT22 5](#_Toc58228790)

[Bảng 2: Sơ đồ nối chân ML8511 6](#_Toc58228791)

[Bảng 3: Sơ đồi nối chân cảm biến bụi mịn 7](#_Toc58228792)

[Bảng 4: Sơ đồ nối chân cảm biến đo pH 8](#_Toc58228793)

[Bảng 5: Sơ đồ nối chân Module Sim800A 9](#_Toc58228794)

[Bảng 6: Tập lệnh AT cấu hình cho Module Sim800A 10](#_Toc58228795)

[Bảng 7: Hàm băm 18](#_Toc58228796)

[Bảng 8: Mô tả thành phần trong mạng Hyperledger Fabric 23](#_Toc58228797)

[Bảng 9: Bảng kết quả đo của trạm quan trắc và các thiết bị chuyên dùng 36](#_Toc58228798)

# DANH SÁCH CÁC HÌNH VẼ, HÌNH ẢNH

[Hình 1: Sơ đồ khối trạm quan trắc 3](#_Toc58228750)

[Hình 2: Board mạch chính 4](#_Toc58228751)

[Hình 3: Cảm biến DHT22 5](#_Toc58228752)

[Hình 4: Cảm biến ML8511 5](file:///C:\Users\Admin\Desktop\DATN\BaoCaoDATN_PhamHoaMau_16T1.docx#_Toc58228753)

[Hình 5: Cảm biến GP2Y10 7](#_Toc58228754)

[Hình 6: Cảm biến đo pH 8](#_Toc58228755)

[Hình 7: Module Sim800A 9](#_Toc58228756)

[Hình 8: Kết nối module sim với board mạch chính 10](#_Toc58228757)

[Hình 9: Thử nghiệm trạng thái Module Sim800A 11](#_Toc58228758)

[Hình 10: Kết nối với MQTT Broker 11](#_Toc58228759)

[Hình 11: Publish dữ liệu đến MQTT broker 11](#_Toc58228760)

[Hình 12: Bộ biến đổi năng lượng mặt trời 12](#_Toc58228761)

[Hình 13: Sơ đồ thuật toán Firmware 13](#_Toc58228762)

[Hình 14: Kiến trúc nguyên khối và kiến trúc microservice 14](#_Toc58228763)

[Hình 15: Docker và microservice 16](#_Toc58228764)

[Hình 16: Công nghệ Blockchain 17](#_Toc58228765)

[Hình 17: Ứng dụng của công nghệ Blockchain 17](#_Toc58228766)

[Hình 18: Ví dụ về hàm băm 18](#_Toc58228767)

[Hình 19: Ứng dụng Blockchain trong giao dịch 19](#_Toc58228768)

[Hình 20: Các phiên bản Blockchain 20](#_Toc58228769)

[Hình 21: Yếu tố cơ bản Bitcoin 20](#_Toc58228770)

[Hình 22: Cơ chế đào coin 21](#_Toc58228771)

[Hình 23: Kiến trúc đơn giản của mạng Hyperledger Fabric 23](#_Toc58228772)

[Hình 24: Vụ nổ Big Bang 24](#_Toc58228773)

[Hình 25: Thêm tổ chức vào mạng 25](#_Toc58228774)

[Hình 26: Consurtium trong mạng Hyperledger Fabric 26](#_Toc58228775)

[Hình 27: Channel cho một consortium 26](#_Toc58228776)

[Hình 28: Peer và Ledger 27](#_Toc58228777)

[Hình 29: Chaincode trong mạng Hyperledger Fabric 28](#_Toc58228778)

[Hình 30: Mạng hyperledger cơ bản 30](#_Toc58228779)

[Hình 31: Mô tả hệ thống quan trắc môi trường 32](#_Toc58228780)

[Hình 32: Sơ đồ mạch nguyên lý 32](#_Toc58228781)

[Hình 33: Kiến trúc microservice 33](#_Toc58228782)

[Hình 34: Ứng dụng công nghệ Blockchain 34](#_Toc58228783)

[Hình 35: Giao diện giám sát và truy xuất dữ liệu quan trắc 35](#_Toc58228784)

[Hình 36: Đồ thị so sánh kết quả đo 38](#_Toc58228785)

[Hình 37: Dữ liệu trong sổ cái của hệ thống Blockchain 38](#_Toc58228786)

[Hình 38: Trạm quan trắc giả mạo 39](#_Toc58228787)

[Hình 39: Xác thực dữ liệu trong Blockchain và InfluxDB 40](#_Toc58228788)

[Hình 40: Dữ liệu được sao lưu về máy từ cơ sở dữ liệu 40](#_Toc58228789)

# DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Từ** | **Viết tắt** | **Diễn giải** |
| API | Application Programming Interface | Giao diện lập trình ứng dụng |
| MQTT | Message Queuing Telemetry Transport | Giao thức truyền thông điệp theo mô hình publish/subscribe |
| SPI | Serial Peripheral Bus | Chuẩn truyền thông nối tiếp tốc độ cao |
| AT | Attention command | Tập lệnh chuẩn được hỗ trợ bởi hầu hết các thiết bị di động |
| HTTP | HyperText Transfer Protocol | Giao thức truyền tải siêu văn bản |
| TSDB | Time Series Database | Cơ sở dữ liệu chuỗi thời gian |

# GIỚI THIỆU ĐỀ TÀI

1. **Lý do chọn đề tài**

Như thực trạng hiện nay của tình hình ô nhiễm môi trường và sự nóng lên toàn cầu trên Thế Giới đang là vấn đề được quan tâm lớn nhất của các chuyên gia trong lĩnh vực khí tượng và tất cả chúng ta. Có thể thấy xu thế đổi mới và hội nhập, Việt Nam đã đạt được rất nhiều thành tựu trong quá trình phát triển, vượt qua tác động suy thoái toàn cầu và duy trì tỷ lệ trăng trưởng kinh tế hằng năm. Tuy nhiên, nước ta đang đối mặt với rất nhiều thách thức trong đó có vấn đề ô nhiễm môi trường. Do đó, các giải pháp quan trắc môi trường để đảm bảo phát triển kinh tế đi đôi với bảo vệ môi trường là vấn đề thực sự cấp thiết.

Thế nhưng, các giải pháp quan trắc môi trường hiện nay phần lớn được thực hiện thủ công hoặc định kỳ 3-6 tháng / lần. Các trạm quan trắc tự đông cũng đã được các nước tài trợ nhưng phần lớn được lắp đặt ở hai thành phố lớn là Hồ Chí Minh và Hà Nội. Tuy nhiên mật độ các trạm quan trắc cũng rất thưa thớt.

Do đó, chúng tôi muốn xây dựng một hệ thống quan trắc môi trường, đảm bảo tính kịp thời dữ liệu (dữ liệu được cập nhập liên tục) và sự minh bạch dữ liệu (dữ liệu không bị thay đổi). Những thông tin về chỉ số môi trường của từng địa phương, từng phu phố sẽ được công khai. Mọi người dân và du khách đều có quyền truy cập, theo dõi để đưa ra những biện pháp phòng tránh, các lộ trình du lịch phù hợp để đảm bảo an toàn cho sức

Vì những lí do trên, em đã quyết định chọn đề tài “**Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường**”làm đề tài tốt nghiệp của mình.

1. **Mục đích của đề tài**

Đề tài “**Xây dựng hệ thống Internet of Things hỗ trợ quan trắc môi trường**” được sinh ra nhằm mục đích để xây dựng một hệ thống cho phép quan trắc môi trường tự động và cập nhật dữ liệu liên tục. Đảm bảo tính minh mạch dữ liệu trước những thế lực thao túng và cung cấp thông tin chính xác, trực quan nhất đến người dân giúp họ nắm được thông tin để chủ động bảo vệ bản thân.

1. **Nội dung của đề tài**

Những nội dung công việc cần thực hiện để hoàn thành đề tài bao gồm:

* Tìm hiểu các cảm biến thường dùng trong quan trắc môi trường.
* Mô phỏng và thiết kế trạm quan trắc.
* Tìm hiểu và xây dựng kiến trúc microservice.
* Tìm hiểu và triển khai mạng Blockchain.
* Tìm hiểu, nghiên cứu và áp dụng các ngôn ngữ, công nghệ như NodeJS, Python, Docker để xây dựng hệ thống.
* Nghiên cứu và ứng dụng InfluxDB làm cơ sở dữ liệu chính cho hệ thống.
* Nghiên cứu và cài đặt Grafana làm giao diện giám sát thông tin từ trạm quan trắc
* Viết báo cáo.

1. **Bố cục của đề tài**

Đề tài được chia làm 4 phần chính:

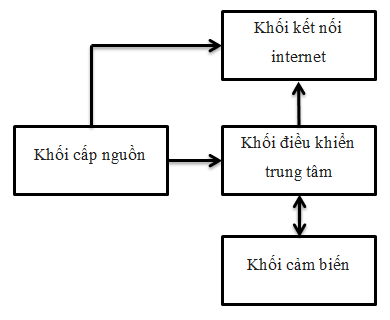
* Phần 1: Tìm hiểu, phân tích các vấn đề của đề tài và đưa ra các giải pháp để giải quyết các vấn đề đó. Tìm hiểu cách kết nối các cảm biến với board mạch chính và giao thức để gửi dữ liệu từ trạm quan về hệ thống.
* Phần 2: Tìm hiểu ưu điểm của kiến trúc microservice so với kiến trúc nguyên khối. Tìm hiểu công nghệ Blockchain và framework có thể áp dụng
* Phần 3: Triển khai hệ thống, thực hiện xây dựng mã nguồn chương trình và chạy thử nghiệm.
* Kết luận: Đưa ra các đánh giá kết quả về những vấn đề đã giải quyết, những vấn đề chưa giải quyết được và các giải pháp để cải tiến, khắc phục những vấn đề đó.

# CHƯƠNG 1: TỔNG QUAN VỀ TRẠM QUAN TRẮC MÔI TRƯỜNG

1. **Phát triển phần cứng**
   1. **Nguyên lý hoạt động**

Board DSPIC30F được kết nối với các cảm biến và Module Sim 800A và cấp nguồn 12V thông qua bộ cung cấp điện năng. Sau khi kết nối thì Board điều khiển hoạt động của các cảm biến thu thập các chỉ số quan trắc về nồng độ pH, cường độ tia UV, khối lượng bụi trong không khí, nhiệt độ và độ ẩm. Board sẽ gửi dữ liệu đo được lên một Micro Cloud đã được xây dựng thông qua Module Sim 800A.

* 1. **Sơ đồ khối**

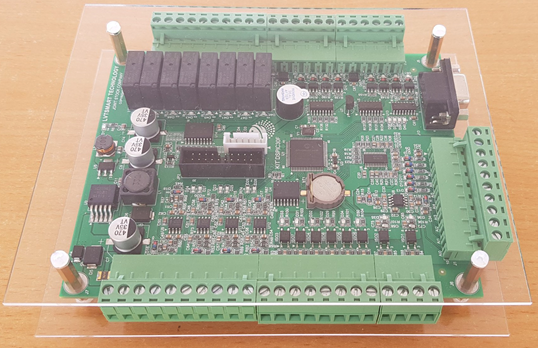


Hình : Sơ đồ khối trạm quan trắc

Chức năng của từng khối:

* Khối cấp nguồn: Chuyển đổi năng lượng của ánh sáng mặt trời thành điện năng sạc vào trong ắc quy. Cung cấp nguồn 12V cho Board mạch chính và Module Sim 800A hoạt động.
* Khối điều khiển trung tâm: Cấp nguồn và điều khiển hoạt động khối cảm biến. Nhận dữ liệu từ khối cảm biến và gửi dữ liệu lên khối kết nối internet.
* Khối cảm biến: Thu thập các thông số về nồng độ pH, cường độ tia UV, khối lượng bụi trong không khí, nhiệt độ, độ ẩm của môi trường xung quanh. Gửi dữ liệu về cho khối điều khiển trung tâm.
* Khối kết nối internet: Nhận dữ liệu từ khối điều khiển trung tâm và gửi dữ liệu lên MircoCloud.
  1. **Linh kiện**
     1. **DSPIC30F Dev Board**

DSPIC30F Dev Board phát triển dựa trên chíp DSPIC30F6014A và một chip ADS1248 để chuyển đổi tín hiệu tương tự sang số 24 bit (ADC) dựa trên SPI giao tiếp.



Hình : Board mạch chính

Vi điều khiển DSPIC30F6014A :

* Tập lệnh cơ bản gồm có 83 lệnh.
* Bộ nhớ chương trình Flash 144 Kbytes.
* Bộ nhớ RAM độ lớn 8 Kbytes.
* Bộ nhớ EEPROM 4 Kbytes.
* Mảng 16 thanh ghi làm việc 16 bits.
* Tốc độ làm việc lên tới 30MIPS.
* Phạm vi điện áp hoạt động rộng (2,5V - 5,5V).
* ADS1248 – bộ chuyển đổi ADC 24-bit:
* Tốc độ dữ liệu lập trình lên tới 2k SPS.
* Bộ ghép kênh tương tự với 8 đầu vào có thể lựa chọn độc lập.
* PGA từ 1 đến 128.

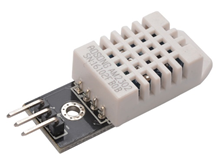
Các cổng vào ra (I/O) và các ngoại vi:

* Dòng ra, vào ở các chân I/O lớn: 25 mA.
* 5 Timer 16-bit, có thể ghép 2 Timer 16-bit thành Timer 32-bit.
* Chức năng Capture 16-bit.
* Các bộ so sánh/PWM 16-bit.
* Module SPI 3 dây (hỗ trợ chế độ Frame).
* Module I2C hỗ trợ chế độ Multi-Master / Slave, địa chỉ 7-bit đến 10-bit.
* UART có khả năng địa chỉ hoá, hỗ trợ bộ đệm FIFO.
* Bộ chuyển đổi tương tự sang số có 16 kênh đầu vào (ADC) 12 bit.
  + 1. **Cảm biến nhiệt độ và độ ẩm**

Cảm biến DHT22 là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp one wire. Bộ tiền xử lý tín hiệu tích hợp trong cảm biến giúp bạn có được dữ liệu chính xác nhất.

Thông số kỹ thuật:

* Điện áp hoạt động: 5VDC.
* Dãi độ ẩm hoạt động: 0% - 100% RH, sai số ±2%RH.
* Dãi nhiệt độ hoạt động: -40°C ~ 80°C, sai số ±0.5°C.



Hình : Cảm biến DHT22

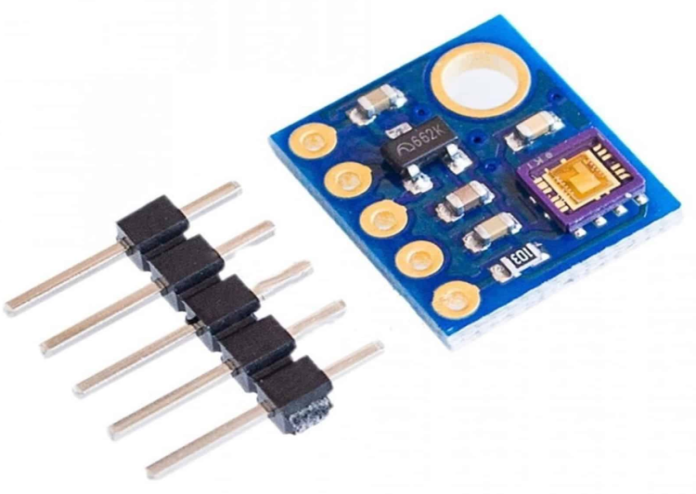
Các chân kết nối với Board :

Bảng : Sơ đồ nối chân cảm biến DHT22

|  |  |
| --- | --- |
| **DHT22** | **BOARD** |
| VCC | +5v |
| GND | GND |
| DOUT | F1 |

Dữ liệu từ cảm biến DHT22 trả về là một chuỗi dài gồm 5 byte:

* 2 byte đầu của độ ẩm.
* 2 byte tiếp theo của nhiệt độ.
* 1 byte cuối cùng của Checksum.
  + 1. **Cảm biến cường độ tia UV**

Cảm biến tia UV sử dụng IC cảm biến từ hãng Lapis cho độ nhạy cao, cảm biến có tích hợp sẵn bộ khuếch đại analog nên có thể lấy trực tiếp dữ liệu Ananog từ cảm biến và cho vào ADC của Vi điều khiển mà không cần thêm các mạch khuếch đại

Hình : Cảm biến ML8511

Thống số kỹ thuật:

* Điện áp cung cấp: 3.3 - 5VDC.
* Cảm biến có độ nhạy cao với tia UV-A và UV-B.
* Tích hợp bộ khuếch đại nội.
* Đầu ra trả dữ liệu Analog.
* Dòng tiêu thụ thấp 300 mA, dòng nghỉ 0.1 mA.

Các chân kết nối chân với Board:

Bảng : Sơ đồ nối chân ML8511

|  |  |
| --- | --- |
| **ML8511** | **BOARD** |
| Vin | +5v |
| EN | +5v |
| Dout | B3 |
| GND | GND |

* + 1. **Cảm biến bụi trong không khí**

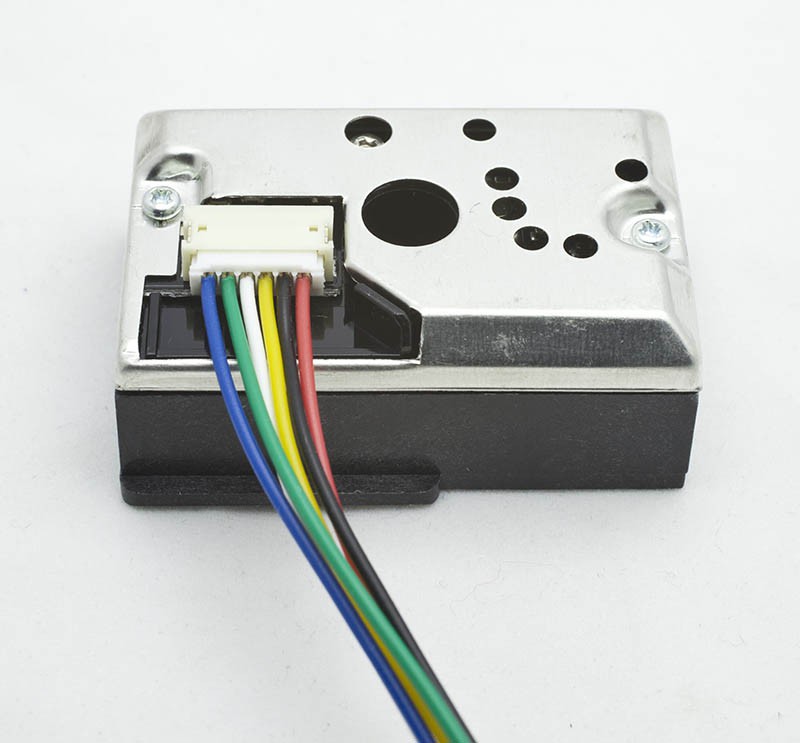
Cảm biến bụi GP2Y10 được sản xuất bởi hãng SHARP, được sử dụng để nhận biết nồng độ bụi PM2.5 trong không khí.

Thông số kỹ thuật:

* Điện áp sử dụng: 5VDC.
* Nhiệt độ hoạt động: -10 – 65oC.
* Định mức dòng: 15mA

Cấu tạo: Gồm 3 phần chính:

* IR LED.
* Phototransistor.
* Amplifier.



Hình : Cảm biến GP2Y10

Các chân kết nối với Board:

Bảng : Sơ đồi nối chân cảm biến bụi mịn

|  |  |
| --- | --- |
| **GP2Y10** | **BOARD** |
| V-LED | +5v(có trở 150 Ohm ở giữa) |
| LED-GND | GND |
| LED | C4 |
| S-GND | GND |
| Vo | AI2 |
| Vcc | +5v |

Nguyên lý hoạt động: Có 2 bộ phận dùng để truyền và nhận hồng ngoại (IR LED và Phototransistor) được đặt chệch gốc với nhau. Khi có bụi bay vào, tia hồng ngoại từ IR LED sẽ bị dội vào Phototransistor, lúc này điện áp từ phototransistor sẽ được đưa đến mạch khuếch đại (Amplifier) và xuất ra chân Vo.

* + 1. **Cảm biến nồng độ pH trong nước**

Gồm có 2 bộ phận chính:

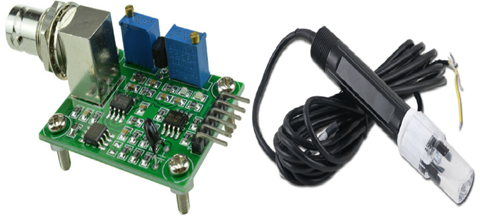
* Module chuyển đổi giá trị pH.
* Đầu dò pH.

Thông số kỹ thuật:

* Module chuyển đổi giá trị pH:
* Điện áp sử dụng: 5±0.2V (AC.DC).
* Dòng điện: 5-10mA.
* Khoảng đo pH: 0-14.
* Nhiệt độ làm việc: -10~50oC (thông thường 20℃ ).
* Độ ẩm làm việc: 95%RF (độ ẩm danh nghĩa 65%RF).
* Thời gian đáp ứng: ≤5s.
* Thời gian ổn định: ≤60s.
* Tuổi thọ: 3 năm.
* Kích thước: 42×32×20mm.
* Cân nặng: 25g
* Tín hiệu ngõ ra: analog.

Đầu dò pH:

* Nhiệt độ hoạt động: 0~60oC.
* Thang đo (pH): 0~14+ Giá trị pH trung bình: 7±0.5.
* Bù nhiệt độ: NTC10K/NTC2.252K/PT100/PT1000.
* Điện cực tham chiếu: điện cực kép Ag/AgCl.
* Chất liệu: PPS/PC.
* Kết nối: sử dụng ren chuẩn 3/4NPT.



Hình : Cảm biến đo pH

Các chân kết nối với Board:

Bảng : Sơ đồ nối chân cảm biến đo pH

|  |  |
| --- | --- |
| **Cảm biến pH** | **BOARD** |
| V+ | +5v |
| GND | GND |
| Po | AI1 |

**Nguyên lý hoạt động của đầu dò pH**: Một điện cực pH được cấu tạo bởi hai loại thủy tinh. Thân điện cực được làm bằng loại thủy tinh không dẫn điện, đầu điện cực thường có dạng hình bầu và cấu tạo bởi loại thủy tinh có công thức gồm các oxit silica, lithium, canxi và các nguyên tố khác cho phép ion lithium xuyên qua.

Cấu trúc của điện cực thủy tinh cho phép ion lithium trao đổi với các ion hydro trong chất lỏng tạo thành lớp thủy hợp. Một điện thế cỡ mV được sinh ra giữa tiết diện của đầu thủy tinh pH với dung dịch lỏng bên ngoài. Độ lớn của điện thế này phụ thuộc vào giá trị pH của dung dịch. Mức điện thế khác nhau tạo ra bởi lớp bên ngoài và lớp thủy hợp bên trong điện cực có thể đo bằng điện cực bạc/bạc cloride. Dung dịch bên trong điện cực thủy tinh có giá trị pH không thay đổi, cho nên điện thế thay đổi chỉ khi giá trị pH của dung dịch bên ngoài thay đổi.

* + 1. **Module Sim800A**

Module Sim 800A có khả năng nhắn tin SMS, nghe, gọi, GPRS, … như một điện thoại nhưng có kích thước nhỏ. Điều khiển module sử dụng bộ tập lệnh AT dễ dàng, chân kết nối dùng rào đực thông dụng (male header) chuẩn 100mil.



Hình : Module Sim800A

Thông số kỹ thuật:

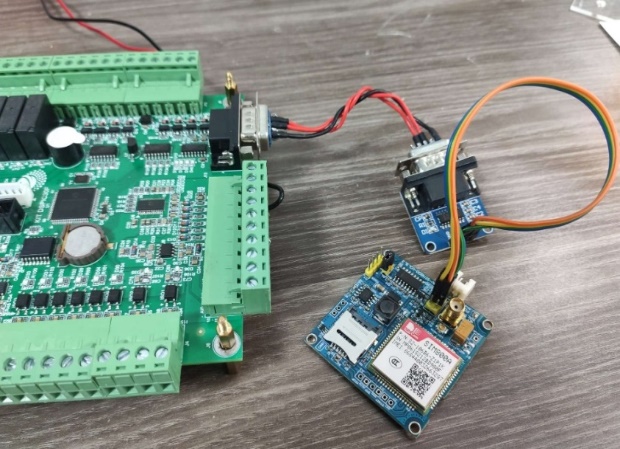
* Điện áp sử dụng: DC5-18V 2A.
* Kết nối với vi điều khiển, hoặc theo chuẩn USB to RS232.
* Kích thước 47x50mm.
* Các tập lệnh sử dụng trong tài liệu của SIM800.
* Nhiệt độ hoạt động: -40 – 85oC.

Các chân kết nối với Board :

Bảng : Sơ đồ nối chân Module Sim800A

|  |  |
| --- | --- |
| **Module Sim800A** | **BOARD** |
| VCC | +5v |
| GND | GND |
| RXD | TXD |
| TXD | RXD |

Để giao tiếp giữa DSPIC30F Dev Board và Module Sim800 sử dụng Max232 để chuyển đổi tín hiệu RS232 sang TTL qua cổng giao tiếp UART2



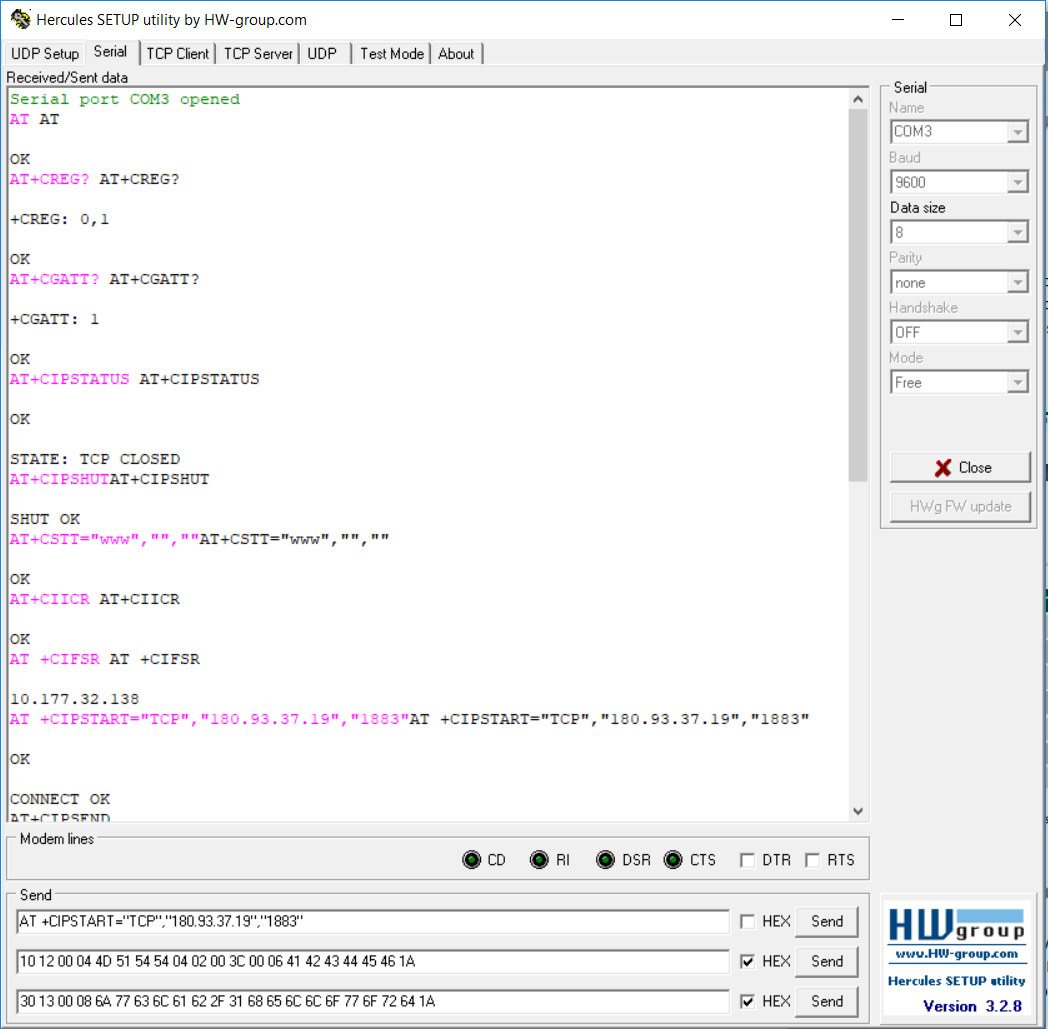
Hình : Kết nối module sim với board mạch chính

Các tập lệnh AT cấu hình cho Module Sim:

Bảng : Tập lệnh AT cấu hình cho Module Sim800A

|  |  |  |
| --- | --- | --- |
| **Tập lệnh** | **Ý nghĩa** | **Trả về** |
| AT | Kiểm tra giao tiếp giữa 2 thiết bị | OK |
| AT+CREG? | Kiểm tra sim đã sẵn sàng và kết nối với mạng | Success Result: +CREG=1,1  OK |
| AT+CGATT? | Kiểm tra xem SIM có truy cập Internet không | Response: +CGATT: 1 |
| AT+CIPSTATUS | Truy vấn cài đặt IP | OK  STATE: IP INITIAL |
| AT+CIPSHUT | Đóng cổng TCP | SHUT OK |
| AT+CSTT="www","","" | Thiết lập APN, tên người dùng và mật khẩu cho PDP. | OK |
| AT+CIICR | Đưa lên mạng wireless. Đảm bảo Module Sim có tín dụng hoặc có gói dữ liệu | OK |
| AT +CIFSR | Trả lại địa chỉ ip cục bộ. |  |
| AT+CIPSTART=”TCP”,”IP”,”port” | Giao thức TCP, địa chỉ ip gửi và port | OK |
| AT+CIPSEND | Gửi dữ liệu qua kết nối TCP hoặc UDP |  |

Màn hình Terminal sau khi nhập tập lệnh AT:



Hình : Thử nghiệm trạng thái Module Sim800A

Kết nối và gửi dữ liệu lên Broker MQTT :



Hình : Kết nối với MQTT Broker



Hình : Publish dữ liệu đến MQTT broker

* + 1. **Bộ chuyển đổi năng lượng mặt trời.**

Gồm có 4 thiết bị:

* Pin mặt trời.
* Bộ chuyển đổi sạc pin mặt trời cho ắc quy.
* Đồng hồ hiển thị năng lượng mặt trời.
* Ắc quy.

Pin mặt trời:

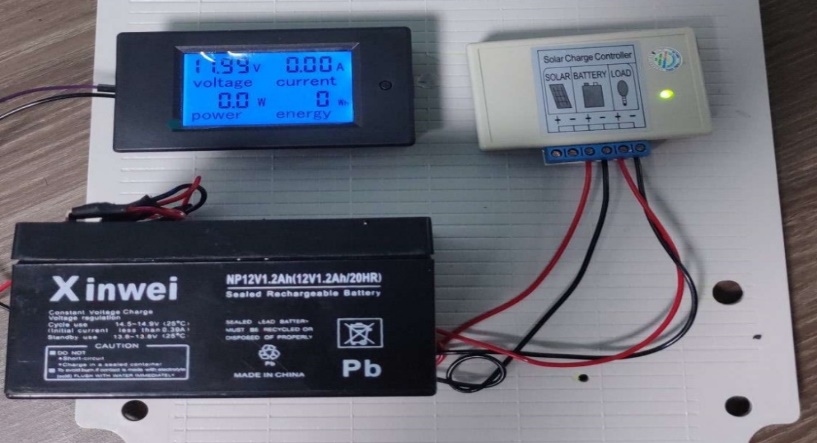
* Sử dụng tấm pin năng 5W/12v.
* Kích thước: 255x192x16mm.
* Điện áp cực đại: 17.9v.
* Dòng cực đại: 0.28 An.
* Độ lệch điện: ± 5%.

Bộ chuyển đổi sạc pin năng lượng mặt trời cho ắc quy:

* Sử dụng cho các tấm pin mặt trời sạc Acquy 12V hoặc Acquy 24V.
* Khi hiệu điện thế của pin năng lượng mặt trời lớn hơn 10v thì bộ sạc hoạt động.
* Khi ánh sáng yếu (<10v) thì led trên bộ sạc sẽ nhấp nháy.
* Đồng hồ đo hiển thị năng lượng mặt trời
* Điện áp đo và hoạt động: 6.5 ~ 100VDC, sai số 0.01.
* Dòng điện đo và hoạt động: 0 ~ 20A, sai số 0.01.
* Công suất đo và hoạt động: 0 ~ 2000W, sai số 0.1.
* Năng lượng đo và hoạt động: 0~9999kWh.
* Màn hình hiển thị: LCD, đèn nền xanh dương, có thể bật tắt đèn nền.
* Kích thước màn hình: 30 x 51mm.
* Kích thước sản phẩm: 84.6 x 49.6 x 24.4mm.

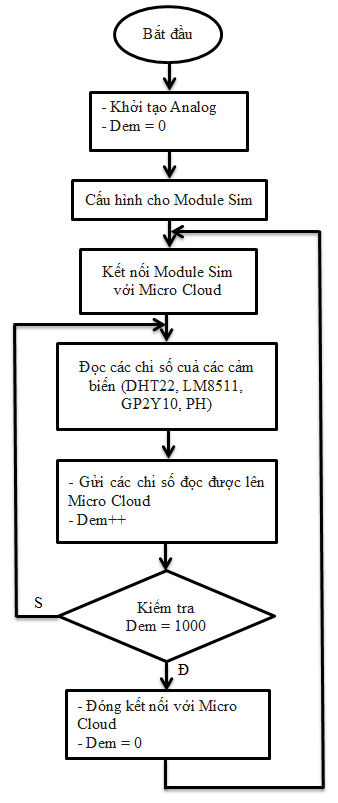
Ắc quy:

* Sử dụng 2 pin 12v 1,2Ah ghép song song lại với nhau.
* Tích hợp Pin năng lượng mặt trời và ắc quy:
* Tấm pin năng lượng mặt trời được nối vào phần SOLAR của bộ sạc.
* Ắc quy được nối vào phần BATTERY của bộ sạc.
* Đồng hồ hiển thị năng lượng mặt trời được nối với LOAD của bộ sạc.
* Từ trên LOAD của bộ sạc được nối ra kết nối với Board và Module Sim.



Hình : Bộ biến đổi năng lượng mặt trời

1. **Phát triển firmware**
   1. **Sơ đồ thuật toán**



Hình : Sơ đồ thuật toán Firmware

* 1. **Phầm mềm hỗ trợ**

Sử dụng ngôn ngữ C hướng cấu trúc để lập trình trên phần mềm CCS C Compiler phiên bản 5.015.

Sử dụng phần mềm Burn–E và mạch nạp Burn–E để nạp code từ phần mềm xuống Board.

Sử dụng chuẩn giao tiếp SPI để kết nối giữa chip ADS1248 với Board.

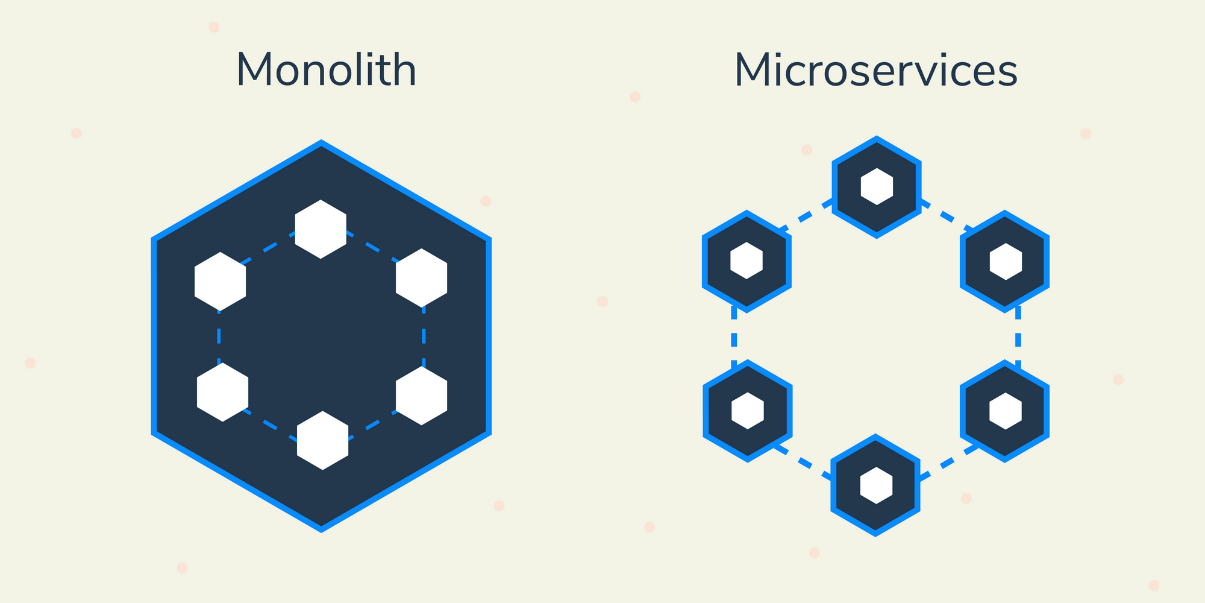
# CHƯƠNG 2: GIỚI THIỆU VỀ KIẾN TRÚC MICROSERVICE

1. **Kiến trúc microservice là gì?**

Không có định nghĩa phổ quát về thuật ngữ microservice. Định nghĩa đơn giản nhất của microservice, còn được gọi là kiến trúc microservice (microservice architecture), là một kiểu kiến ​​trúc ứng dụng sử dụng các dịch vụ được liên kết lỏng lẻo. Các dịch vụ này có thể được phát triển, triển khai và duy trì độc lập.

Chúng hoạt động với tốc độ nhanh hơn và đáng tin cậy hơn nhiều so với các ứng dụng nguyên khối, phức tạp truyền thống. Sử dụng kiến ​​trúc microservice, một tổ chức có kích thước bất kỳ có thể phát triển các ngăn xếp công nghệ phù hợp với khả năng của họ.

Có nhiều lợi ích hữu hình khi sử dụng microservice, chúng ta sẽ thảo luận sau, nhưng vẫn còn một số tranh cãi về việc liệu các công ty có nên chuyển từ kiến ​​trúc nguyên khối sang kiến ​​trúc microservice hay không. Hãy xem xét sự khác biệt giữa hai kiến trúc để hiểu cuộc tranh luận.



Hình : Kiến trúc nguyên khối và kiến trúc microservice

**Kiến trúc nguyên khối** (monolithic) là cách truyền thống để xây dựng và triển khai các ứng dụng. Cấu trúc này dựa trên khái niệm về một đơn vị duy nhất, không thể chia tách, bao gồm phía máy chủ, phía máy khách và cơ sở dữ liệu. Tất cả các khía cạnh được thống nhất và quản lý như một đơn vị và cơ sở mã. Điều này có nghĩa là mọi cập nhật phải được thực hiện cho cùng một cơ sở mã, vì vậy toàn bộ ngăn xếp phải được thay đổi. Khi các ứng dụng nguyên khối có quy mô, chúng có thể trở nên khá phức tạp, do đó sự phát triển tổng thể thường dài hơn.

Ngược lại, **kiến trúc microservice** chia nhỏ khối đó thành các đơn vị độc lập có chức năng như các dịch vụ riêng biệt. Điều này có nghĩa là mọi dịch vụ đều có logic và cơ sở mã riêng. Chúng giao tiếp với nhau thông qua các API (giao diện lập trình ứng dụng). Vậy, nên chọn kiến ​​trúc nào?

**Chọn kiến ​​trúc nguyên khối**

* **Nếu công ty của bạn là một nhóm nhỏ**:Sử dụng kiến trúc này, bạn không phải đối phó với sự phức tạp của việc triển khai kiến ​​trúc microservice.
* **Nếu bạn muốn khởi động nhanh hơn**:Kiến trúc nguyên khối nhỏ đòi hỏi ít thời gian hơn để khởi động. Hệ thống này sẽ cần nhiều thời gian hơn để cập nhật hệ thống của bạn, nhưng việc khởi chạy ban đầu nhanh hơn.

**Chọn kiến ​​trúc microservice**

* **Nếu bạn muốn phát triển một ứng dụng có khả năng mở rộng hơn**:Mở rộng một kiến ​​trúc microservice dễ dàng hơn nhiều. Các chức năng và mô-đun mới có thể được thêm vào rất dễ dàng và nhanh chóng.
* **Nếu công ty của bạn lớn hơn hoặc có kế hoạch phát triển**:Sử dụng microservice rất tốt cho một công ty có kế hoạch phát triển, vì kiến ​​trúc microservice có khả năng mở rộng và dễ dàng tùy chỉnh hơn theo thời gian.

1. **Ưu điểm và nhược điểm của microservice**

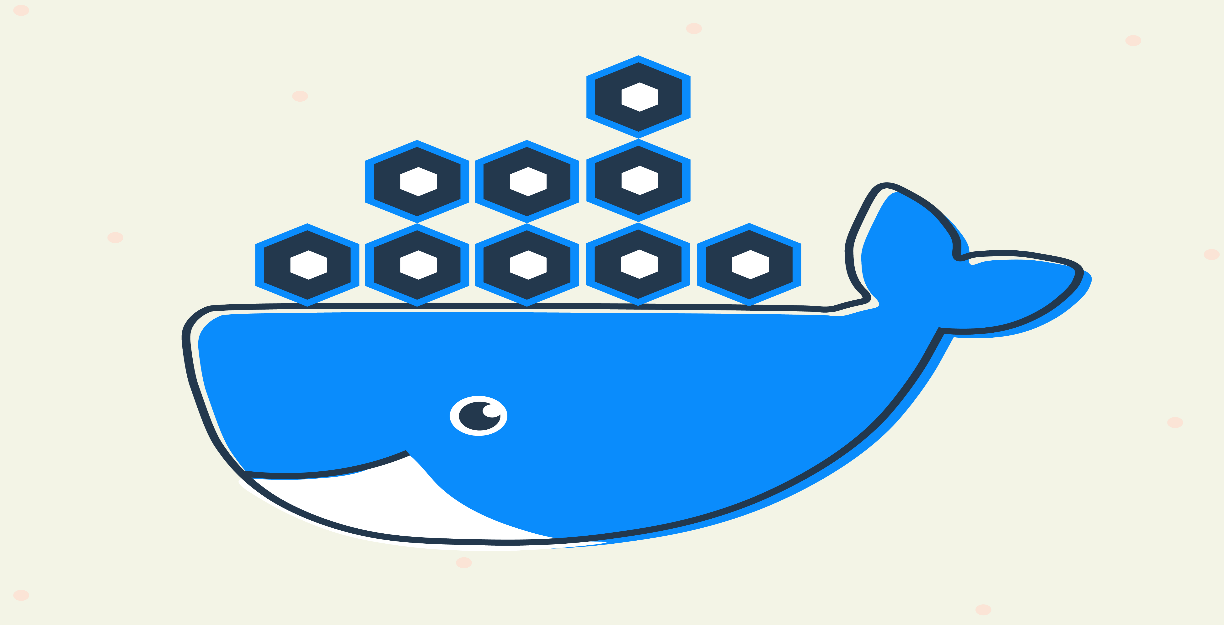
**Ưu điểm của microservice**

* **Cải thiện khả năng mở rộng và năng suất:** Các nhóm lớn thường phải làm việc cùng nhau trong các dự án phức tạp. Với microservice, các dự án có thể được chia thành các đơn vị nhỏ hơn, độc lập.Điều này có nghĩa là các nhóm có thể hành động độc lập về logic nghiệp vụ, giúp giảm thiểu sự phối hợp và nỗ lực. Trên hết, các nhóm chịu trách nhiệm cho từng microservice có thể đưa ra quyết định công nghệ của riêng họ tùy thuộc vào nhu cầu của họ. Ví dụ: cấu trúc bên trong của mỗi đơn vị hoặc vùng chứa không quan trọng miễn là giao diện hoạt động chính xác. Do đó, bất kỳ ngôn ngữ lập trình nào cũng có thể được sử dụng để viết microservice, vì vậy nhóm chịu trách nhiệm có thể chọn ngôn ngữ tốt nhất cho đồng đội của họ.
* **Tích hợp tốt với các hệ thống cũ:** Hệ thống nguyên khối rất khó để duy trì. Nhiều hệ thống cũ có cấu trúc kém, thử nghiệm kém hoặc phụ thuộc vào các công nghệ lỗi thời.May mắn thay, microservice có thể làm việc cùng với các hệ thống cũ để cải thiện mã và thay thế các phần cũ của hệ thống.Tích hợp rất dễ dàng và có thể giải quyết nhiều vấn đề khiến các hệ thống nguyên khối trở thành quá khứ.
* **Phát triển bền vững:** Kiến trúc microservice tạo ra các hệ thống có thể duy trì trong thời gian dài do các bộ phận khác nhau có thể thay thế.Điều này có nghĩa là một microservice có thể dễ dàng được viết lại mà không ảnh hưởng đến toàn bộ hệ thống. Miễn là sự phụ thuộc giữa các microservice được quản lý phù hợp, các thay đổi có thể dễ dàng được thực hiện để tối ưu hóa nhu cầu và hiệu suất của nhóm.
* **Chức năng chéo:** Microservice là tốt nhất cho các nhóm phân phối. Nếu bạn có các nhóm trên toàn thế giới hoặc các bộ phận khác nhau, microservice cấp các quyền tự do cần thiết và linh hoạt để hoạt động tự chủ. Các quyết định kỹ thuật có thể được đưa ra nhanh chóng, tích hợp với các dịch vụ khác trong nháy mắt. Chức năng chéo chưa bao giờ được dễ dàng hơn.

**Nhược điểm của microservice**

* **Triển khai đòi hỏi nhiều nỗ lực hơn:** Hoạt động của một hệ thống microservice thường đòi hỏi nhiều nỗ lực hơn, vì có nhiều đơn vị có thể triển khai hơn mà mỗi đơn vị phải được triển khai và giám sát. Thay đổi giao diện phải được thực hiện để có thể triển khai độc lập các microservice riêng lẻ.
* **Kiểm thử phải độc lập:** Vì tất cả các microservice phải được kiểm thử cùng nhau, một microservice có thể làm gián đoạn thử nghiệm và ngăn chặn việc triển khai các microservice khác. Có nhiều giao diện để kiểm tra và kiểm tra phải độc lập cho cả hai mặt của giao diện.
* **Khó thay đổi nhiều microservice:** Những thay đổi ảnh hưởng đến nhiều microservice có thể khó thực hiện hơn. Trong một hệ thống microservice, các thay đổi đòi hỏi một số triển khai phối hợp.

1. **Microservice và Docker**



Hình : Docker và microservice

[Docker](https://comdy.vn/docker/) và [microservice](https://comdy.vn/microservice/) gần như đồng nghĩa. Các dịch vụ phải được triển khai riêng biệt, các đơn vị độc lập có thể mở rộng. Nhưng, nếu bạn tạo nhiều microservice cho ứng dụng của bạn thì sao? Docker là một giải pháp nhẹ để triển khai microservice.

Một microservice có thể được đóng gói vào một image Docker và được phân lập như một container Docker. Bằng cách này, bạn có thể xây dựng một ứng dụng độc lập với môi trường máy chủ của mình. Thay vì có một máy ảo hoàn chỉnh của riêng mình, các container Docker chia sẻ kernel của hệ điều hành trên máy chủ Docker. Các tiến trình từ các container xuất hiện trong bảng tiến trình của hệ điều hành mà các container Docker đang chạy.

Để sử dụng Docker với microservice, bạn cần tạo image Docker thông qua tập tin có tên là Dockerfile. Dockerfile rất dễ viết, vì vậy phần mềm có thể xuất bản dễ dàng.

Một hệ thống microservice điển hình chứa nhiều container Docker. Phối hợp một hệ thống nhiều container Docker yêu cầu cấu hình cho mạng ảo.

Các container phải có khả năng tìm thấy nhau để liên lạc. Môi trường Docker Compose có thể liên hệ với một máy chủ khác thông qua một liên kết, cung cấp một hệ thống khám phá dịch vụ.

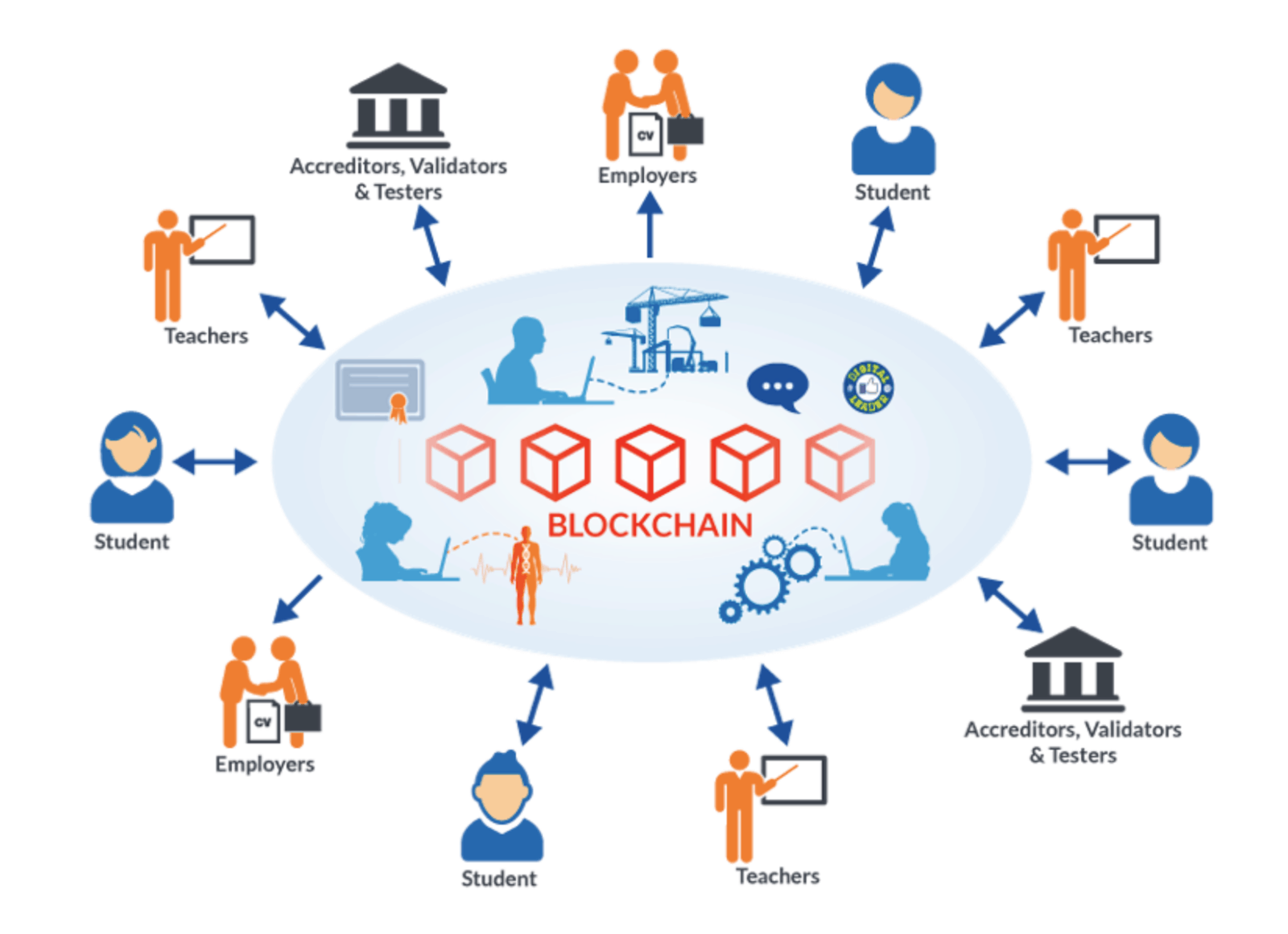
# CHƯƠNG 3: TÌM HIỂU CÔNG NGHỆ BLOCKCHAIN

1. **Tổng quan về Blockchain**



Hình : Công nghệ Blockchain

Công nghệ Blockchain (chuỗi khối) thường được liên kết với Bitcoin và các loại tiền điện tử khác, nhưng đó chỉ là phần nổi trong mảng băng chìm. Ngày nay, Blockchain đã có thể chuyển sang các ngành công nghiệp quan trọng, từ chăm sóc sức khỏe sang chính trị.

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/e8dbb6a4-8c46-4b0a-8f02-bfa776e06b6f_Screen20Shot202018-08-1220at205.31.1520AM.png)

Hình : Ứng dụng của công nghệ Blockchain

Công nghệ Blockchain không đơn giản, ý tưởng cơ bản của là một cơ sở dữ liệu mà cộng đồng sẽ xác thực thay vì xác thực tập trung như truyền thống ở bên thứ ba (ngân hàng, nhà nước). Nó lưu trữ tất cả các hồ sơ, lịch sử, mọi thứ trên mạng lưới phân tán và mọi người đều có quyền ngang hàng như nhau thay vì dựa vào một máy chủ cụ thể như ngân hàng (chính phủ có thể thao túng).

Mỗi khối đại diện cho một số giao dịch, và thành phần chuỗi của nhóm liên kết bằng hàm băm (Hash function). Khi các bản ghi được tạo, chúng được xác nhận bởi mạng lưới máy tính phân tán và được ghép nối với mục nhập trước đó trong chuỗi, từ đó tạo thành các chuỗi khối (gọi là Blockchain).

Blockchain được liên kết với nhau bằng mã hóa và các trường thời gian, mỗi khối thông tin đều chứa thông tin về thời gian khởi tạo và được liên kết với khối trước đó, kèm một mã thời gian và dữ liệu giao dịch. Một khi dữ liệu đã được xác nhận thì không có cách nào chỉnh sửa nó nữa – làm cho nó vô cùng minh bạch và không thể bị làm giả vì không ai có thể quay lại và thay đổi thứ gì.

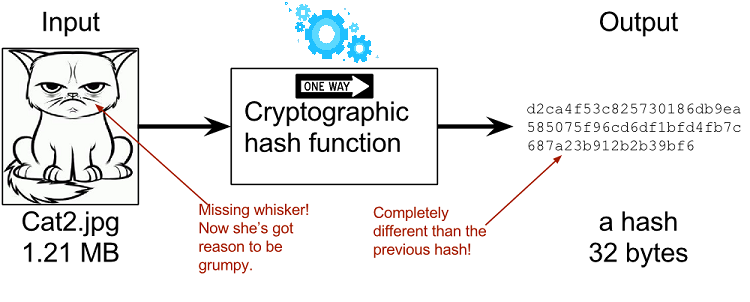
* 1. **Hàm băm**

Hàm băm có chức năng chuyển dữ liệu có độ dài bất kỳ thành một chuỗi ký tự hoặc số duy nhất có độ dài cố định. Ví dụ:

Bảng : Hàm băm

|  |  |
| --- | --- |
| the quick brown fox jumps over the lazy dog | 77add1d5f41223d5582fca736a5cb335 |
| the quick | e8394495128fcd958836523af1601f00 |
| the quikk | d58f6b5a222de711eb8cc4580b8459b0 |

Bất kỳ sự thay đổi nào cũng sẽ dẫn đến thay đổi hoàn toàn giá trị của hàm băm.

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/Rosenbaum_CHaB_02.png)

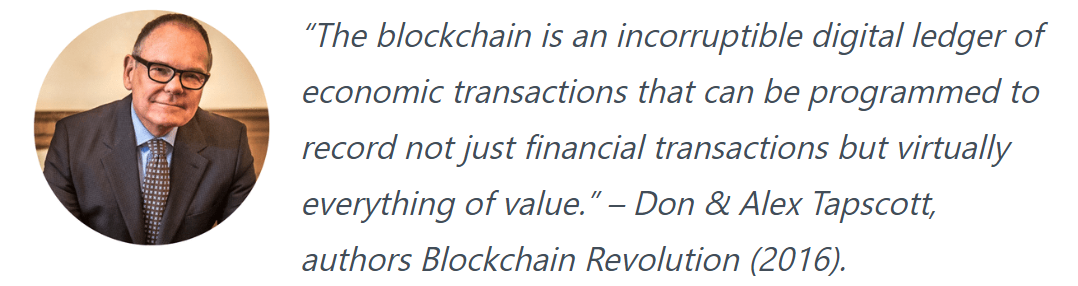
Hình : Ví dụ về hàm băm

* 1. **Khối (Block) và dữ liệu khối**

Mỗi khối là tập hợp các lịch sử của giao dịch mà các thợ đào xác nhận và liên kết lên chuỗi. Mỗi khối đều có Mã băm, Dữ liệu giao dịch, Thời gian, Mã băm của khối trước, khối sau liên kết với khối trước bằng mã băm cứ vậy tiếp diễn.

Các khối thông tin hoạt động độc lạp và có thể mở thêm ra theo thời gian, khi có một khối thông tin được ghi vào hệ thống thì không thể chỉnh sửa được gì và chỉ được thêm vào khi có sự xác nhận của tất cả mọi người.

* 1. **Blockchain ứng dụng trong giao dịch**

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/Transact.png)

Hình : Ứng dụng Blockchain trong giao dịch

Trong cuốn sách Blockchain Revolution, Don&Alex Tapscott đã nhận định “Blockchain là một sổ cái điện tử không thể bị phá hỏng, và ghi lại tất không chỉ những giao dịch tài chính mà tất cả mọi thứ có giá trị”.

Trên Blockchain không tồn tại phí giao dịch (chỉ có chi phí cơ sở hạ tầng), Blockchain tạo ra cách để truyền thông tin từ A đến B một cách đơn giản nhưng khéo léo hoàn toàn tự động và an toàn. Một bên tham gia giao dịch bắt đầu quá trình bằng cách tạo các khối, khối này được xác minh bởi hàng ngàn, có khi hàng triệu máy tính được phân phối trên thế giới. Sau khi được xác minh khối sẽ thêm vào chuỗi, tạo ra một bản ghi duy nhất mà lịch sử của nó không thể bị làm sai lệch (nếu muốn làm sai bản ghi duy nhất này bạn phải làm sai tất cả các chuỗi đồng nghĩa đánh sập cả mạng lưới Blockchain, tức là đánh sập hơn 51% các khối trong chuỗi, điều này không thể xảy ra được).

Trong thị trường tiền điện tử thì mọi người đều hiểu Blockchain như một cuốn sổ cái, nơi mà mọi hoạt động của nó đều được ghi lại và được giám sát chặt chẽ, công khai và minh bạch.

* 1. **Nút của Blockchain**

Mỗi nút (node) trong Blockchain đều đang lưu trữ một bản sao của sổ cái, bởi vậy mọi người đều biết số tài sản điện tử của bạn là bao nhiêu, nhưng chỉ có mình bạn mới có thể sử dụng được nó. Hệ thống không theo dõi số dư của bạn, mà nó ghi lại tất cả các giao dịch.

Bạn có thể thắc mắc làm sao để nó xác thực các giao dịch đầu vào? Thực tế các nút sẽ kiểm tra tất cả các giao dịch trước đó có liên quan để tham chiếu lịch sử giao dịch từ đó xác nhận các giao dịch có hợp lệ hay không.

1. **Nguyên lý hoạt động của Blockchain**

Bạn cứ tưởng tượng rằng một bảng tính được nhân đôi hàng ngàn lần trên một mạng máy tính, sau đó hãy tưởng tượng rằng mạng này được thiết kế để cập nhật cái bảng tính này liên tục, nhưng mỗi khi nó cập nhật phải được đa số mọi người trong mạng chấp thuận là bạn có thể hiểu được cơ bản về Blockchain.

Để một khối (block) được thêm vào chuỗi khi có giao dịch phải:

Xác minh giao dịch. Mọi thông tin liên quan đến giao dịch như tên giao dịch, thời gian, địa điểm, số tiền giao dịch, người tham gia, (tất cả các dữ liệu liên quan đến giao dịch) đều phải được ghi lại. Rồi sau đó phải có đủ xác nhận từ mọi người trên mạng lưới.

Kế tiếp là được hàm băm (hash function) chuyển đổi. Chỉ khi được chuyển đổi thành các ký tự và số qua hàm băm thì mới đóng khối và thêm vào chuỗi.

Công nghệ Blockchain cho phép thực hiện giao dịch mà không cần một bên trung gian nào cả.

* 1. **Các phiên bản của Blockchain**

**Blockchain 1.0: Tiền tệ**

Ý tưởng tạo ra tiền như là một phần thưởng thông qua việc giải đố các bài toán được Hal Finney giới thiệu lần đầu vào năm 2005, người đã tọa ra khái niệm đầu tiên cho tiền điện tử (việc triển khai của sổ cái công nghệ phân tán). Sổ cái này cho phép các giao dịch tài chính dựa trên công nghệ Blockchain, Bitcoin chính là ứng dụng đầu tiên của BlockChain, Bitcoin được coi như “Tiền của Internet”.

**Blockchain 2.0: Hợp đồng thông minh (Smart Contracts)**

Các vấn đề xảy ra với Bitcoin là khải thác quá lãng phí và thiếu khả năng mở rộng mạng lưới. Để khắc phục các vấn đề này, Blockchain mở rộng ra khỏi lĩnh vực tiền điện tử phát triển chương trình mới là “Hợp đồng thông minh”. Chúng là các chương trình máy tính trong Blockchain, nó tự động thực thi và kiểm tra các điều khoản, tốc độ xử lý của nó nhanh hơn rất nhiều so với Bitcoin, và ví dụ rõ nhất chính là Ethereum.

**Blockchain 3.0: Dapps (Decentralized Applications – Ứng dụng phi tập trung)**

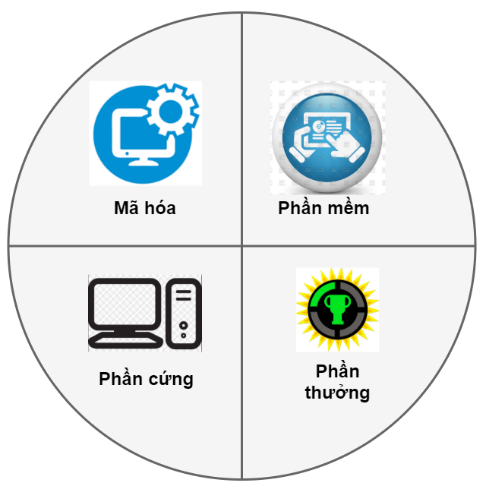
Dapps được sử dụng lưu trữ phi tập trung và công khai, mã của nó chạy trên một mạng ngang hàng (Peer-to-Peer). Một Dapps có thể có giao diện người dùng được lưu trữ trên các kho lưu trữ phi tập trung và có thể được viết bằng bất kỳ ngôn ngữ nào.

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/Verson.png)

Hình : Các phiên bản Blockchain

* 1. **Cơ bản về Bitcoin**

Các yếu tố cơ bản về Bitcoin:

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/btc.png)

Hình : Yếu tố cơ bản Bitcoin

Phần mềm: về cơ bản Bitcoin là một phần mềm cốt lõi để xác định Bitcoin được chuyển như thế nào trong giao dịch. Nó xác định các quy tắc để xác minh, cái gì hợp lệ, cái gì không, phần mềm Bitcoin này luôn được vận hành 24/7.

Mã hóa: phần mềm chính là cốt lõi của Bitcoin, nó sử dụng mã để điều chỉnh việc chuyển đổi Bitcoin giữa các bên, cũng như việc tạo ra đơn vị Bitcoin mới.

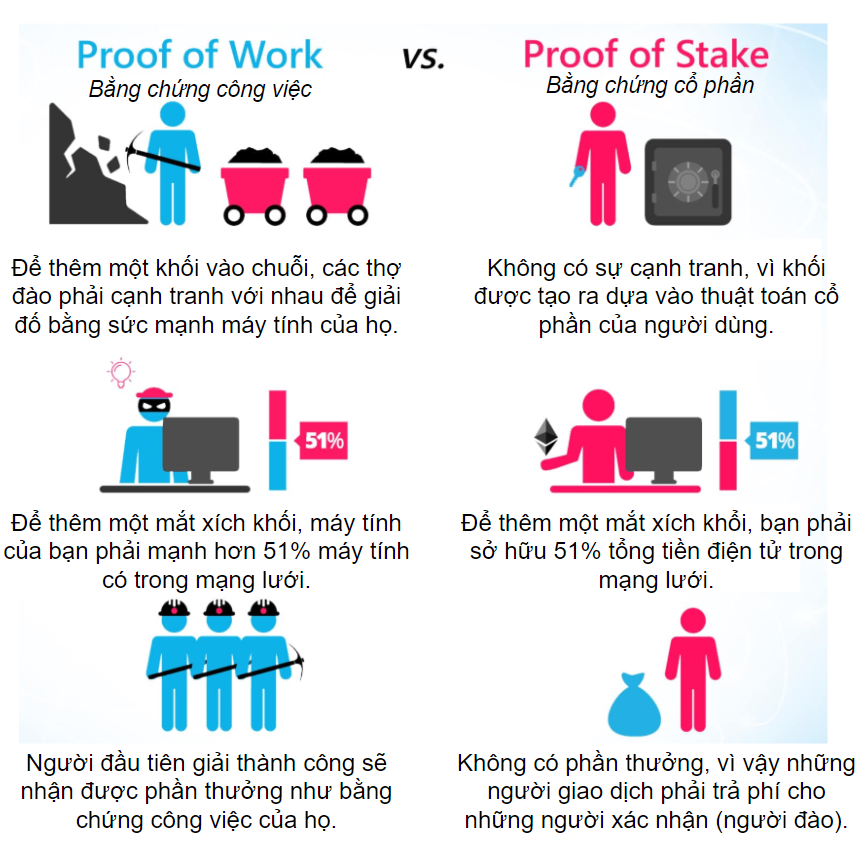
Phần cứng: để chạy và giải mã, nó cần phải có phần cứng, phần cứng này bao gồm hàng hàng, triệu và có thể hơn nữa các thợ mỏ trên khắp thế giới chạy máy tính. Phần cứng này được thiết kế để xác thực khối thông qua các hàm băm (Hash function).

Đào Bitcoin và lý thuyết trò chơi: những người khai tác là những người thực sự tham gia vào quá trình hoạt động của Bitcoin, bạn có thể tưởng tượng, Bitcoin đưa ra một thử thách mỗi 10 phút về việc cố gắng giải mã đoạn dữ liệu bằng các hàm băm và xác thực nó, sau khi tạo khối thành công, khối sẽ được đưa vào chuỗi, và những thợ đào sẽ nhận được phần thưởng chính là các đồng Bitcoin mới cho công sức của họ.

* 1. **Cơ chế đào coin và hiểu sâu hơn về Blockchain**

Mining (đào): hay còn gọi là đào coin, là hoạt động sử dụng sức mạnh tính toán để xử lý các thuật toán liên quan đến mạng lưới Blockchain (Proof-of-Work bằng chứng công việc).

Staking: là hoạt động mà ở đó người khai thác thay vì dùng thuật toán để khai thác thì người đó có thể dùng số lượng tiền mà người đó nắm giữ để đào. Điều này có nghĩa là ai nắm giữ càng nhiều số lượng coin thì càng có sức mạnh để khai thác.

[](https://www.fiahub.com/blog/wp-content/uploads/2019/07/Prooof.png)

Hình : Cơ chế đào coin

**Proof of Work (bằng chứng công việc)**

Là thuật toán đồng thuận ban đầu của mạng Blockchain, thuật toán này được sử dụng để xác nhận giao dịch và tạo một khối mới cho chuỗi. Trong thuật toán này, những Miner (thợ đào) cạnh tranh với nhau để hoàn thành giao dịch, quá trình cạnh tranh này được gọi là khai thác (đào), ngay sau khi thợ đào tạo thành công một khối hợp lệ họ sẽ nhận được phần thưởng, ứng dụng đầu tiên của nó chính là Bitcoin.

Quá trình PoW có thể là quá trình ngẫu nhiên với xác suất thấp, trong quá trình này, rất nhiều thử nghiệm và sai sót được yêu cầu nhưng chỉ khi nó hợp lệ thì mới tạo ra được khối mới.

**Proof of Stake (bằng chứng cổ phần)**

Một người có thể đào hay xác nhận giao dịch khối theo số lượng tiền mà người đó nắm giữ, PoS được tạo ra như một giải pháp để thay thế PoW. Trong PoS thay vì các thợ đào phải xác nhận thì các người dùng trong hệ sinh thái sẽ xác nhận. Theo đó, người dùng sẽ đặt cược vào các khối mà họ cảm thấy sẽ được thêm vào chuỗi, khi mỗi khối được thêm vào chuỗi thì họ sẽ nhận thưởng tương ứng với số cổ phần của họ.

Ví dụ: Bạn sẽ chọn 1 trong 10 người có số tài sản lớn nhất, họ sẽ đại diện và sẽ tham gia xác nhận khối, nếu họ xác nhận khối thành công thì bạn và những người khác đã chọn họ sẽ nhận được những phần thưởng tương ứng với phần trăm.

Proof of Stake sẽ tiết kiệm năng lượng hơn, an toàn hơn vì nếu một hacker muốn tấn công anh ta phải có ít nhất 51% tổng số tiền của thị trường.

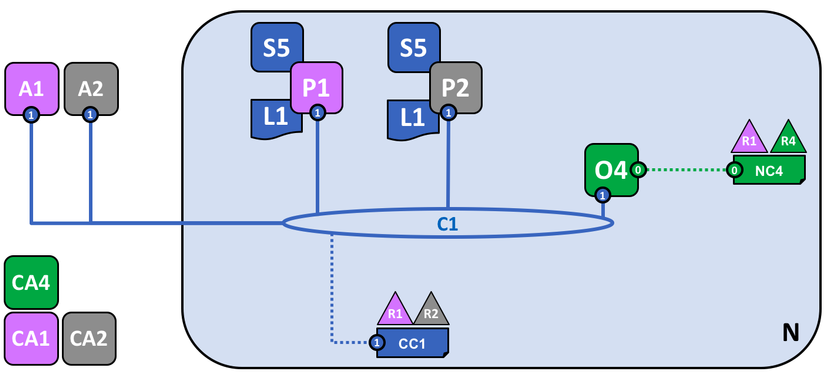
1. **Triển khai hệ thống Blockchain hỗ trợ quan trắc môi trường**
   1. **Giới thiệu về Hyperledger Fabric**

**Hyperedger Fabric** là một trong 5 Framework về Blockchain nằm trong chiến lược **Hyperledger Umbrella** của **Linux Foundation** gồm : Hyperledger Indy, Hyperledger Fabric, Hyperledger Iroha, Hyperledger Sawtooth, Hyperledger Burror. Điều đặc biệt là Hyperledger Fabric được contributed bởi ông lớn **IBM**.

Hyperledger Fabric có **modularity** (tính mô đun) khá cao nên nó cho phép các Doanh nghiệp dễ dàng **plug and play** để xây dựng một ứng dụng Private Blockchain phù hợp các yêu cầu nghiệp vụ của mình.

Phiên bản mới nhất của Hyperledger Fabric tại thời điểm viết bài này là 2.2.0 LTS

* 1. **Kiến trúc đơn giản của một mạng Hyperledger Fabric**



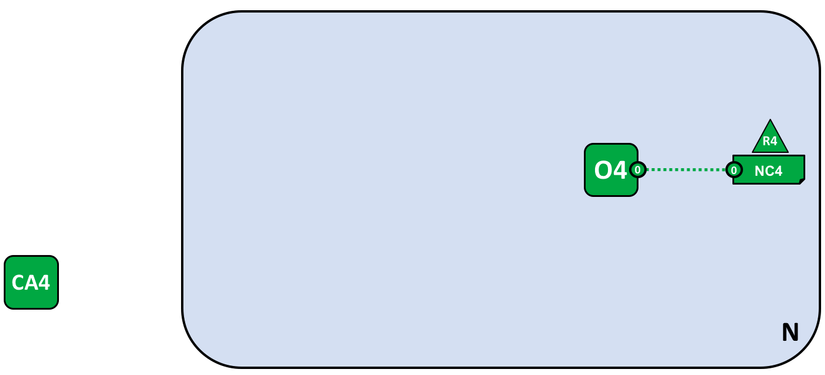
Hình : Kiến trúc đơn giản của mạng Hyperledger Fabric

Bảng : Mô tả thành phần trong mạng Hyperledger Fabric

|  |  |
| --- | --- |
| **Kí hiệu** | **Mô tả** |
| **N** (Network) | Mạng. |
| **NC** (Network Configuration) | Cấu hình của mạng |
| **C**: Channel | tập hợp các tổ chức có vai trò nhất định trong cùng một quy trình kinh doanh. Ví dụ, trong một channel về mua bán xe hơi sẽ gồm có 2 tổ chức là : Nhà sản xuất xe hơi, Nhà phân phối xe hơi. |
| **CC** (Channel Configuration) | Cấu hình của kênh |
| **R** (Organization) | Tổ chức |
| **O** (Orderer Node) | Nếu như trong Publuc Blockchain, tất cả các node của mạng đều tham gia vào quá trình đồng thuận, thì ở Hyperledger Fabric chỉ có Orderer tham gia vào quá trình đó. |
| **P** (Peer) | điểm tương tác giữa các thành viên trong tổ chức tương ứng với kênh, mọi hành động của người dùng đều phải đi qua peer |
| **S(**SmartContract-Chaincode) | được cài đặt trên kênh, định nghĩa rõ các struct, các hành động mà người dùng có thể thực hiện để tương tác trạng thái của struct được lưu trong sổ cái. |
| **L**: Ledger ( Sổ cái ), | lưu trữ trạng thái của các đối tượng |
| **CA** : Certificate Authority | phát hành identity cho người dùng hoặc node của tổ chức tương ứng. Ví dụ, người dùng A là thành viên của Tổ chức R1, khi muốn tham gia vào mạng thì sẽ gửi yêu cầu đến CA1, sau đó CA1 sẽ tạo ra một identity gồm private-key, public-key và các đặc tính liên quan khác, sau đó trả về cho người dùng A, từ đó về sau A dùng identity đó để thực hiện các tương tác với mạng, mạng sẽ tự động biết đó là người dùng A đến từ tổ chức R1. |
| **A**: Application | ứng dụng hay giao diện (web, mobile app ) giúp người dùng tương tác với hệ thống dễ dàng hơn. |

**Quy trình xây dựng mạng**

**Bước 1. Vụ nổ Big Bang**

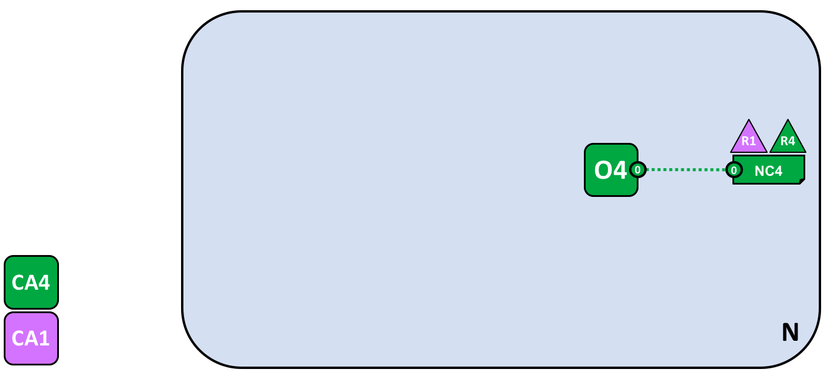


Hình : Vụ nổ Big Bang

Hình trên là sơ khởi của một mạng N. Gồm có một Orderer Node O4, trên đó chạy một dịch vụ có tên là Ordering Services, tổ chức R4 nắm giữ quyền quản trị mạng N và thông tin này được lưu trong cấu hình mạng NC4, node CA4 có nhiệm vụ phát hành identity cho người dùng hoặc peer hoặc application đến từ tổ chức R4.

Như vậy, đùng một phát chả hiểu từ đâu ra xuất hiện R4 cung cấp một Orderer O4 cho mạng, mọi hành động sau này như thêm tổ chức vào vào mạng, thêm kênh, cài đặt chaincode cho kênh, khởi tạo chaincode, yêu cầu thực thi chaincode,... đều phải đi qua Orderer O4 này ( bạn sẽ thấy rõ điều này ở các bài sau ). Và trong Hyperledger Fabric, tất cả các hành động trên đều là transaction ( giao dịch ).

**Bước 2. Thêm một tổ chức quản trị**



Hình : Thêm tổ chức vào mạng

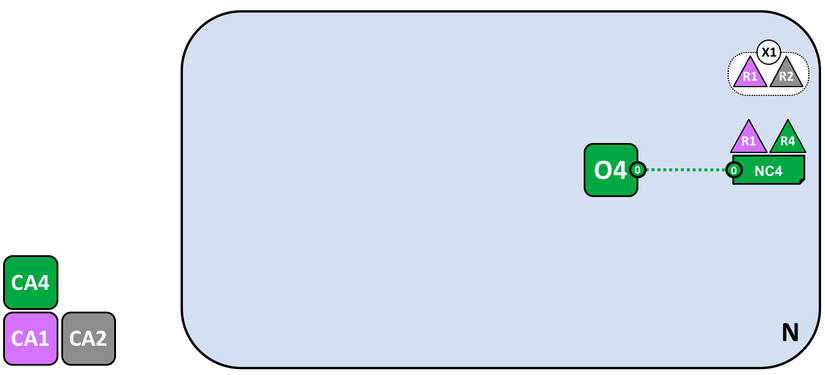
NC4 ban đầu được cấu hình để chỉ cho phép người dùng R4 có quyền quản trị trên mạng. Trong bước này, mình sẽ thêm một tổ chức R1 vào mạng và cho phép R1 có quyền quản trị như R4:

* Tổ chức R4 cập nhật cấu hình mạng NC4 để thêm tổ chức R1 làm quản trị viên. Sau thời điểm này, R1 và R4 có quyền ngang nhau đối với cấu hình mạng.
* Ta thấy CA1 cũng được thêm vào, CA1 sẽ cung cấp indentity cho người dùng của tổ chức R1. Sau thời điểm này cả người dùng từ R1 và người dùng từ R4 đều có quyền quản trị mạng.
* Mặc dù O4 đang chạy trên một cơ sở hạ tầng nào đó của R4 nhưng R1 cũng có quyền như R4 đối với O4.

Mình sẽ trình bày về Ordering service ở những bài sau, bây giờ bạn chỉ cần hiểu O4 như một điểm quản trị tất cả các hoạt động của mọi thành phần trong mạng.

**Bước 3. Định nghĩa 1 consortium**

Hiện tại mạng có thể được quản lý bởi R1 và R4, và có rất ít hành động mà có thể thực hiện được đối với mạng. Để có thể mapping các hoạt động kinh doanh vào mạng thì điều đầu tiên chúng ta cần làm là định nghĩa một **consortium** ( nhóm - tập đoàn ). Từ này có nghĩa đen là một nhóm các tổ chực cùng nằm trong một hoạt động kinh doanh, ví dụ Tổ chức sản xuất ô tô, Tổ chức phân phối ô tô được sản xuất bởi Tổ chức sản xuất đến tay Người tiêu dùng.

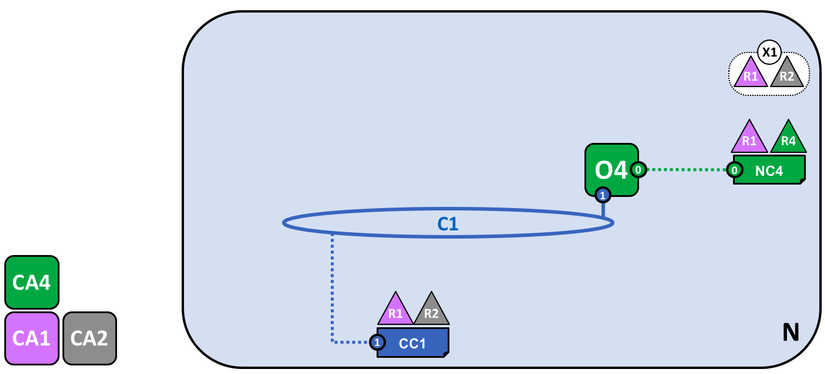


Hình : Consurtium trong mạng Hyperledger Fabric

Một quản trị viên mạng (R1 hoặc R4) định nghĩa một consortium X1 có chứa hai thành viên, R1 và R2. Định nghĩa của consortium này được lưu trữ trong cấu hình mạng NC4 và sẽ được sử dụng ở giai đoạn phát triển mạng tiếp theo. CA2 là Cơ quan cấp identity cho người dùng, node, application đến từ tổ chức R2 này. Một consortium có số lượng tổ chức tùy ý, ở đây mình sử dụng trường hợp đơn giản nhất là 2.

**Bước 4. Tạo một channel cho một consortium**

Kênh là một phương tiện truyền thông tin mà thông qua đó các thành viên của một consortium có thể giao tiếp với nhau. Có thể có nhiều consortium và nhiều kênh trong một mạng, nhưng mỗi consortium chỉ có một kênh.



Hình : Channel cho một consortium

Một kênh C1 đã được tạo cho consortium X1. Cấu hình của kênh C1 được lưu trong cấu hình kênh CC1, tách biệt hoàn toàn với cấu hình mạng NC4. CC1 được quản lý bởi R1 và R2, 2 tổ chức này có quyền ngang nhau đối với C1. R4 không có quyền gì trong CC1.

Kênh C1 cung cấp một cơ chế liên lạc riêng cho các tổ chức trong X1. Chúng ta có thể thấy kênh C1 mới chỉ được kết nối với Ordering O4. Trong bước tiếp theo, chúng ta sẽ kết nối các thành phần như Application và Peer.

Mặc dù kênh C1 là một phần của mạng N, nhưng nó tách biệt hoàn toàn với N. Cũng lưu ý rằng tổ chức R4 không nằm trong kênh này - kênh này chỉ dành cho xử lý giao dịch giữa R1 và R2. Trong bước trước, chúng ta đã thấy cách R4 đã cấp quyền quản trị mạng cho R1, rồi R1 tạo ra một consortium. Hiểu ngầm rằng R4 cũng đã cấp quyền cho R1 tạo kênh! Trong sơ đồ này, có thể là tổ chức R1 hoặc R4 đã tạo ra kênh C1. Một lần nữa, lưu ý rằng một kênh có thể có bất kỳ số lượng tổ chức nào được kết nối với nó - mình lấy ví dụ đơn giản nhất là 2.

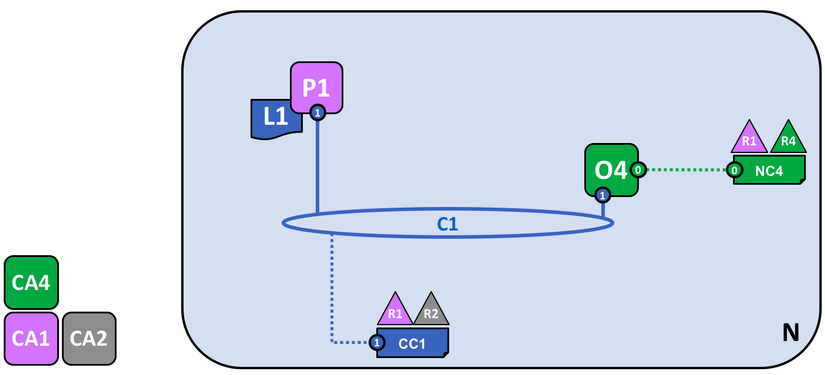
Cấu hình kênh CC1 chứa các quy định về quyền mà R1 và R2 có trên kênh C1 - và như chúng ta thấy R4 không có quyền gì trong kênh này cả. R4 chỉ có thể tương tác với C1 nếu chúng được thêm bởi R1 hoặc R2 vào cấu hình kênh CC1. R4 cũng không thể tự thêm mình vào kênh C1 - nó phải và chỉ được thực hiện vởi R1 và R2.

Đến đây ta có thể thấy rằng tính riêng tư của Hyperledger Fabric đến từ Kênh. Hyperledger Fabric rất mạnh về vấn đề này, vì nó cho phép các tổ chức chia sẻ cơ sở hạ tầng nhưng lại giữ được sự riêng tư của mình.

Mọi cập nhật cho cấu hình mạng NC4 từ thời điểm này trở đi sẽ không có ảnh hưởng trực tiếp đến cấu hình kênh CC1; ví dụ: nếu định nghĩa consortium X1 bị thay đổi, nó sẽ không ảnh hưởng đến các thành viên của kênh C1. Do đó, các kênh rất hữu ích vì chúng cho phép liên lạc riêng giữa các tổ chức cấu thành kênh. Hơn nữa, dữ liệu trong một kênh được cách ly hoàn toàn với phần còn lại của mạng, cũng như các kênh khác.

**Bước 5. Peer và Ledger**

Bây giờ, ta sẽ xem cách sử dụng kênh để kết nối các tổ chức lại với nhau.



Hình : Peer và Ledger

Một peer P1 đã tham gia kênh C1. Mỗi Kênh sẽ chỉ có một sổ cái duy nhất, mỗi Peer sẽ lưu trữ 1 bản sao của sổ cái này để người dùng từ tổ chức tương ứng truy cập. Chẳng hạn như P1 của tổ chức R1 lưu trữ trữ bản sao của sổ cái L1 để người dùng từ tổ chức R1 truy cập.

L1 được **physically hosted** trên peer P1, nhưng **logically hosted** trên kênh C1

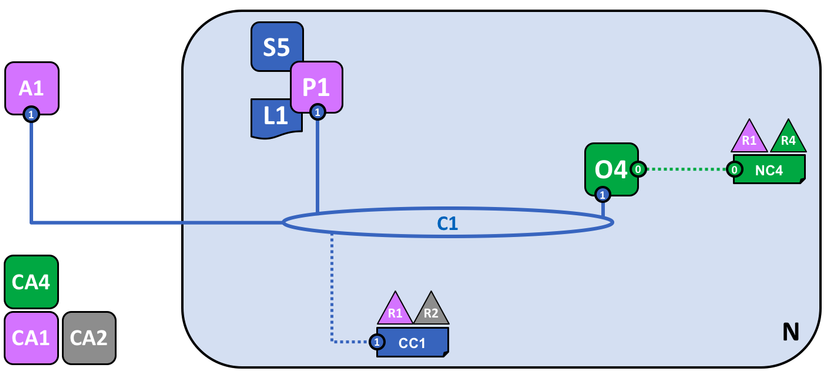
Lúc này, P1 và O4 có thể giao tiếp với nhau thông qua kênh C1.

Mọi thành phần từ người dùng đến peer đều cần có 1 identity. Vì thế P1 là cũng có một identity (kiểu X.509 - chưa cần quan tâm X.509 là gì đâu ) do CA1 cấp, xác định P1 thuộc tổ chức R1.

Khi P1 được start, nó có thể tham gia kênh C1 bằng cách gửi yêu cầu tham gia O4. Khi O4 nhận được yêu cầu tham gia này, nó sử dụng cấu hình kênh CC1 để xác định quyền của P1 trên kênh này. Ví dụ, CC1 xác định xem P1 có thể đọc và / hoặc ghi thông tin vào sổ cái L1 hay không.

**Bước 6. Application và Smart Contract ( Chaincode)**

Bây giờ kênh C1 có một sổ cái trên đó, chúng ta có thể bắt đầu kết nối các application để sử dụng một nghiệp vụ kinh doanh được định nghĩa trong smart contract.



Hình : Chaincode trong mạng Hyperledger Fabric

Một smart contract S5 đã được cài đặt trên P1. Application A1 của tổ chức R1 có thể sử dụng S5 để truy cập vào sổ cái L1 thông qua peer P1. Trong kênh lúc này có A1, P1 và O4.

Cũng giống như các peer, orderer và người dùng, một application sẽ có một identity được liên kết với tổ chức tương ứng. Chẳng hạn A1 có một identity được CA1 cung cấp để xác định A1 thuộc về R1.

Bây giờ có vẻ như A1 có thể truy cập vào sổ cái L1 trực tiếp thông qua P1, nhưng trên thực tế, tất cả quyền truy cập được quản lý thông qua Smart Contract S5. Hiểu đơn giản là S5 định nghĩa tất cả các use case truy cập vào sổ cái L1; S5 cung cấp một tập hợp các cách xác định rõ ràng mà theo đó sổ cái L1 có thể được truy vấn hoặc cập nhật hoặc cập nhật như thế nào, bởi ai. Nói tóm lại, ứng dụng khách A1 phải thông qua Smart Contract S5 để tương tác với sổ cái L1!

Smart Contract có thể được tạo bởi các nhà phát triển ứng dụng trong mỗi tổ chức để thực hiện quy trình kinh doanh được chia sẻ bởi các thành viên của consortium.

Một kênh có thể có nhiều Smart Contract.

**Bước 6.1 Cài một smart contract**

Sau khi smart contract S5 được code xong, quản trị viên trong tổ chức R1 phải cài đặt nó vào peer P1. Đây là một hoạt động đơn giản; sau đó, P1 có thể thấy logic triển khai của S5 - code mà nó sử dụng để truy cập vào sổ cái L1.

Khi một tổ chức có nhiều peer trên một kênh, họ có thể chọn các peer mà họ muốn để cài đặt smart contract; họ không cần phải cài đặt một smart contarct trên mọi peer.

**Bước 6.2 Khởi tạo một smart contract**

Để các thành phần khác được kết nối với kênh C1 biết về smart contract vừa được cài đặt; ta phải khởi tạo nó trên kênh C1. Trong ví dụ này, chỉ có một peer P1, một quản trị viên trong tổ chức R1 phải khởi tạo S5 trên kênh C1 bằng cách sử dụng P1. Sau khi khởi tạo, mọi thành phần trên kênh C1 đều biết về sự tồn tại của S5; điều đó có nghĩa là bây giờ S5 có thể được gọi bởi ứng dụng khách A1.

Lưu ý rằng mặc dù mọi thành phần trên kênh hiện có thể truy cập S5, nhưng chúng không thể thấy logic chương trình của S5. Điều này vẫn được giữ riêng tư đối với những peer đã cài đặt nó; trong ví dụ này có nghĩa là P1. Về mặt khái niệm, điều này có nghĩa là chỉ có interface của smart contract được khởi tạo. Và, cài đặt một smart contract hiểu đơn giản là nó được physically hosted trên một peer, trong khi việc khởi tạo smart contract hiểu là nó được logically hosted trên kênh.

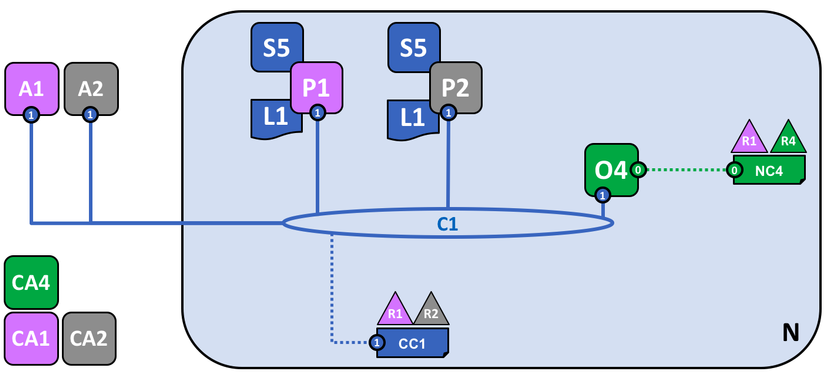
**Endorsement policy ( chính sách chứng thực )**

Phần quan trọng nhất của thông tin phải đượctại thời điểm khởi tạo là một chính sách chứng thực. Nó mô tả các tổ chức nào phải phê duyệt các giao dịch trước khi chúng được các tổ chức khác chấp nhận vào bản sao của sổ cái. ĐỊnh nghĩa kiểu như R1 AND R2 hoặc R1 OR R2.

**Invoking Smart Contract ( gọi hợp đồng thông minh )**

Khi một smart contract được cài đặt trên một peer và được khởi tạo trên một kênh, nó có thể được gọi bởi một application. Các application thực hiện điều này bằng cách gửi đề xuất giao dịch cho các peer thuộc sở hữu của các tổ chức được chỉ định bởi chính sách chứng thực. Đề xuất giao dịch đóng vai trò là tham số đầu vào cho smart contract ( tên function, tham số đầu vào của function đó ), sử dụng nó để tạo ra phản hồi giao dịch được chứng thực, được trả lại bởi peer cho application. Mình sẽ trình bày về transaction flow trong bài sau.

**Bước 7. Network completed**



Hình : Mạng hyperledger cơ bản

Việc thêm các peer P2 và A2, CA2, cài đặt Smart contract cho P2 tương tự như đối với R1.

1. **Ưu điểm của hệ thống Blockchain**
   1. **Tiết kiệm chi phí**

Tiết kiệm chi phí là ưu điểm dễ nhận thấy nhất khi ứng dụng công nghệ Blockchain. Một phần là nó cắt giảm được các quy trình xác minh không cần thiết, giảm thiểu các lỗi và một phần là nó giảm tải việc lưu trữ bằng giấy truyền thống.

* 1. **Tính bảo mật cao**

Dữ liệu một khi được cập nhật vào hệ thống thì rất khó giả mạo và chỉnh sửa, bởi nó sẽ được chia sẻ và được xác nhận bởi hàng triệu máy tính trong mạng lưới.

* 1. **Dữ liệu mang tính bất biến**

Trong mạng Blockchain, nếu như một máy tính bị sập thì dữ liệu trên đó hoàn toàn không bị mất bởi các máy tính khác trong hệ thống đó đã sở hữu bản sao cuốn sổ cái.

* 1. **Giao dịch xuyên biên giới**

Nếu như các giao dịch trước kia bị hạn chế bởi múi giờ và cần phải có sự xác nhận của tất cả các bên thì Blockchain hạn chế sự tham gia của con người vào quá trình xử lý, điều này tạo điều kiện cho các giao dịch xuyên biên giới.

* 1. **Đảm bảo sự minh bạch**

Bạn không bao giờ có thể giả mạo trên mạng Blockchain nhờ sự minh bạch và khả năng kiểm toán của công nghệ này. Vậy nên rất nhiều tổ chức đã tận dụng ưu điểm này để chống rửa tiền.

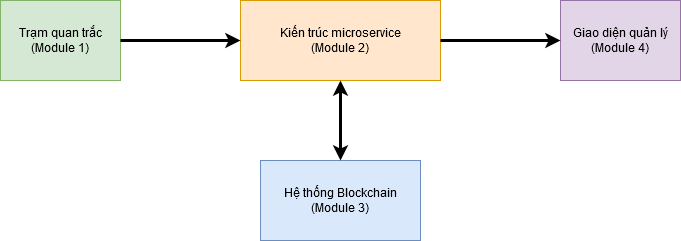
* 1. **Giảm thiểu sự chậm trễ khi xử lý giao dịch**

Hạn chế việc thanh toán bồi hoàn.Dù chỉ là một vài trong rất nhiều những ưu điểm của công nghệ Blockchain, chúng ta cũng có thể nhận thấy tiềm năng to lớn của công nghệ này với cuộc sống

# CHƯƠNG 4: TRIỀN KHAI THỰC NGHIỆM

1. **Mô tả hệ thống**

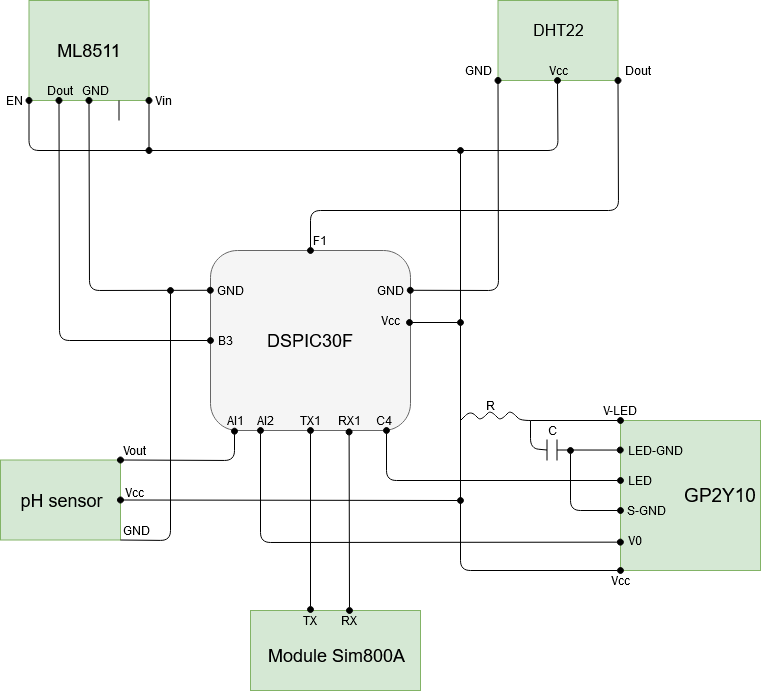
Hệ thống của chúng tôi bao gồm bốn mô-đun: Trạm giám sát, kiến trúc microservice, hệ thống blockchain và giao diện quản lí

****

Hình : Mô tả hệ thống quan trắc môi trường

1. **Trạm quan trắc môi trường**

Trong mô-đun này (Hình 32), chúng tôi thiết kế và xây dựng các trạm quan trắc môi trường sử dụng bo mạch DSPIC30F6014A (bo mạch chủ), DHT22 (cảm biến nhiệt độ và độ ẩm), ML8511 (đo cường độ tia cực tím), GP2Y10 (đo độ bụi trong không khí – PM 2.5), cảm biến pH để đo pH và mô-đun sim800A để gửi dữ liệu đã thu thập đến các Microservice (Mô-đun 2).

****

Hình : Sơ đồ mạch nguyên lý

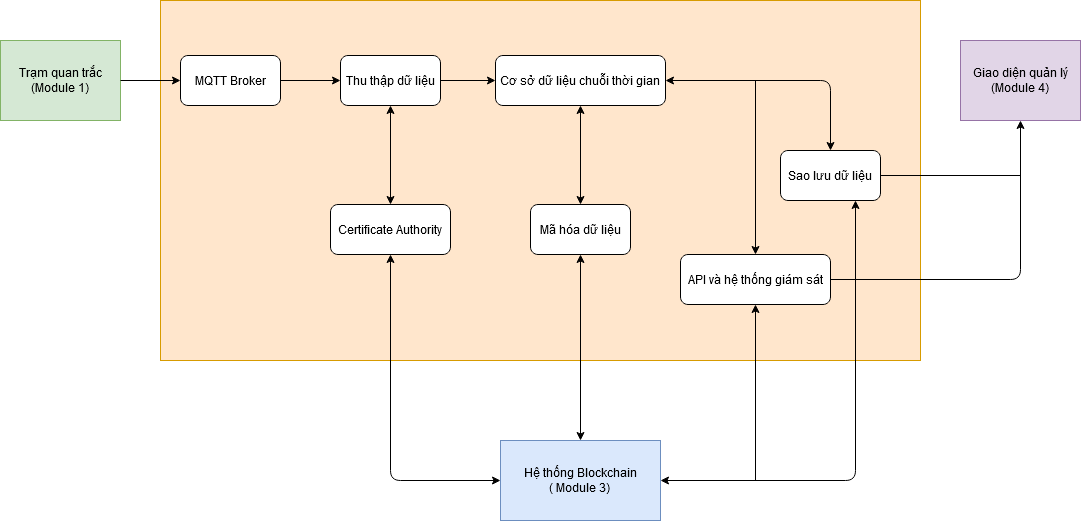
1. **Kiến trúc microservice**

Microservice chia một phần mềm thành các dịch vụ nhỏ hơn có thể được triển khai trên các máy chủ khác nhau. Mỗi dịch vụ sẽ xử lý từng phần của công việc và chúng được kết nối thông qua nhiều giao thức khác nhau, chẳng hạn như HTTP, SOA, socket, Message queue (Active MQ, Kafka), để truyền dữ liệu. Chúng tôi sử dụng kiến ​​trúc microservice thay vì kiến ​​trúc nguyên khối để tận dụng tính khả thi, khả năng mở rộng và ứng dụng quy mô lớn. Mô-đun này (Hình 3) được sử dụng để xử lý dữ liệu nhận được từ Mô-đun 1, bao gồm các thành phần sau:

MQTT broker giao tiếp với các trạm giám sát thông qua Phép đo từ xa hàng đợi tin nhắn Giao thức truyền tải (MQTT). Nó là một giao thức truyền thông điệp với mô hình xuất bản / đăng ký. Dựa trên các chủ đề, MQTT broker có thể nhận dữ liệu được gửi từ các trạm giám sát.

Bộ sưu tập dữ liệu đăng ký dữ liệu thời gian thực từ MQTT broker, và sau đó dữ liệu của các trạm hợp lệ được phép lưu trữ trong cơ sở dữ liệu chuỗi thời gian. Một trạm được coi là hợp lệ nếu tổ chức đã tồn tại trong hệ thống và tên trạm đã được đăng ký. Tất cả thông tin trạm và lịch sử đăng ký được lưu trữ trong hệ thống blockchain.

Cơ sở dữ liệu chuỗi thời gian (TSDB) là cơ sở dữ liệu được tối ưu hóa cho dữ liệu chuỗi thời gian hoặc dấu thời gian. Dữ liệu chuỗi thời gian có thể là dữ liệu máy chủ, dữ liệu mạng, dữ liệu cảm biến và các giao dịch trong thị trường. Trong ứng dụng này, chúng tôi sử dụng cơ sở dữ liệu InﬂuxDB để lưu trữ dữ liệu của các trạm quan trắc.

****

Hình : Kiến trúc microservice

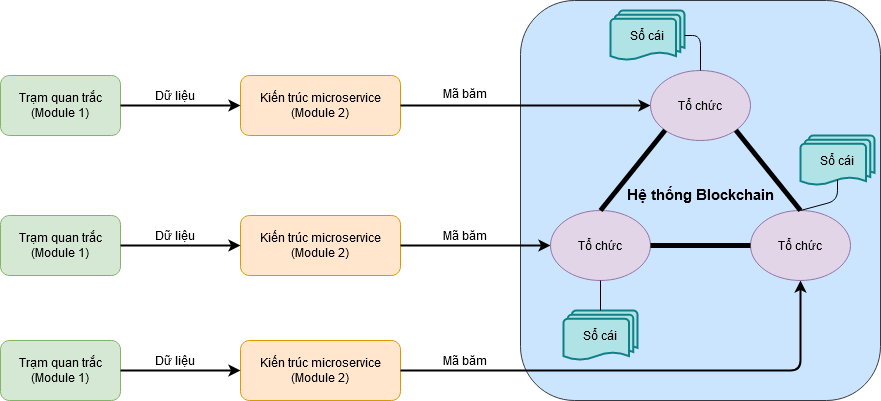
Vì InﬂuxDB lưu trữ dữ liệu theo dấu thời gian và lập chỉ mục theo các trường thẻ, dung lượng lưu trữ nhỏ và tốc độ truy vấn rất cao. Hơn nữa, In ﬂ uxDB cho phép các plugin hỗ trợ các giao thức nhập dữ liệu khác như Graphite, Collectd và OpenTSDB.

* Certifcate Authority cho phép đăng ký tài khoản người dùng và quản trị viên mới trong hệ thống blockchain. Khi người dùng đăng ký tài khoản, Cơ quan chứng nhận tạo chứng chỉ để xác thực trong mạng blockchain như khóa riêng tư, cert, pem.
* Thuật toán mã hóa MD5 thực hiện mã hóa dữ liệu trong InﬂuxDB. Sau đó, dữ liệu được lưu trữ trong hệ thống blockchain để xác minh nguồn gốc của dữ liệu, bảo vệ tính toàn vẹn và ngăn chặn sự thay đổi nội dung của dữ liệu.
* Giao diện lập trình ứng dụng (API) và hệ thống giám sát hỗ trợ Mô-đun 4 để tạo giao diện web và ứng dụng di động. Ngoài ra, chúng tôi phát triển một chức năng sao lưu (định dạng .txt) để lưu trữ cục bộ, thống kê hoặc khôi phục dữ liệu khi cơ sở dữ liệu chuỗi thời gian gặp sự cố. Tuy nhiên, dữ liệu cơ sở dữ liệu trước khi sao lưu cần được xác thực với hệ thống blockchain thông qua mã băm để ngăn chặn việc sửa đổi dữ liệu trái phép.

1. **Hệ thống Blockchain**

Hệ thống blockchain được xây dựng để bảo vệ tính toàn vẹn và minh bạch của dữ liệu của các thiết bị IoT trong việc giám sát môi trường. Chúng tôi tạo các nút trong mạng blockchain theo mô hình phi tập trung, nghĩa là mỗi nút được quản lý bởi từng tổ chức và được xây dựng trên hệ thống đám mây của tổ chức đó. Cả hai đều giúp tối ưu hóa việc quản trị trong mạng blockchain đồng thời giảm thiểu nguy cơ bị tấn công mạng.

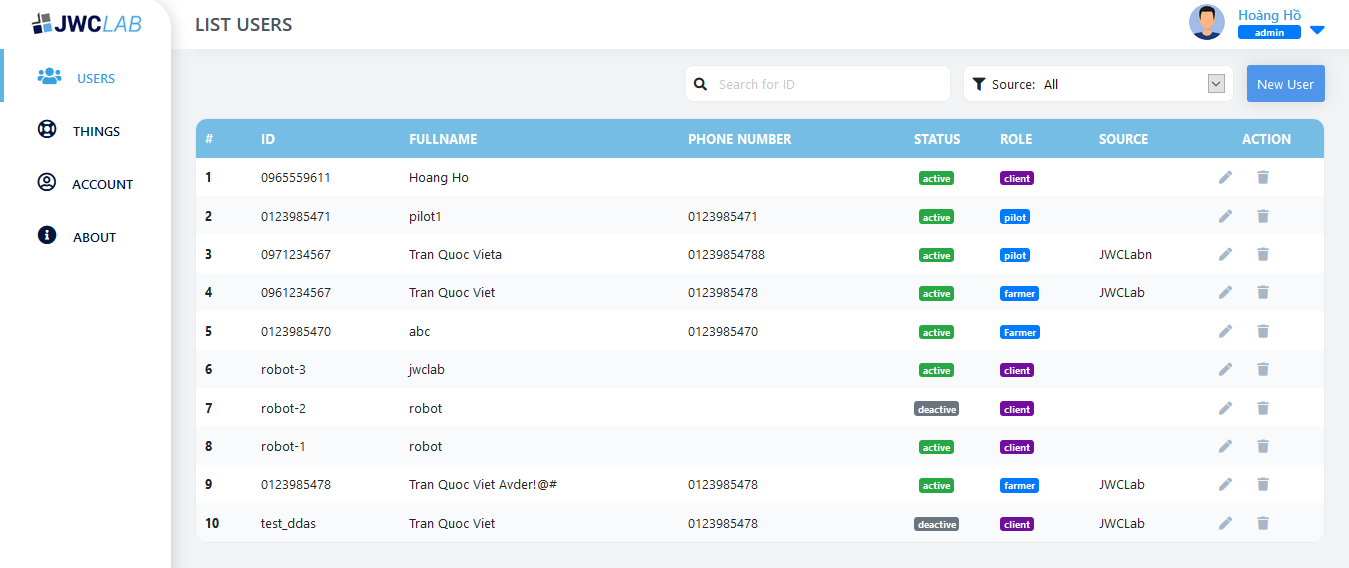
Để xây dựng một hệ thống blockchain, chúng tôi sử dụng Hyperledger Fabric Framework (Hình 34). Mô-đun này tạo các nút trong mạng blockchain và cho phép các nút này ghi dữ liệu vào blockchain với nội dung dữ liệu. Nội dung dữ liệu bao gồm các trường: DateTime (thời gian để ghi vào blockchain), phạm vi thời gian (thời gian bắt đầu và kết thúc của khối dữ liệu trong cơ sở dữ liệu chuỗi thời gian và mã băm (băm MD5 của khối dữ liệu được lấy đúng lúc range). Dữ liệu blockchain là bất biến, do đó các mã băm được lưu trữ sẽ được đảm bảo không thay đổi. Nếu có một thay đổi nhỏ trong cơ sở dữ liệu chuỗi thời gian sẽ gây ra sự thay đổi mã băm. Việc phát hiện các cuộc tấn công trở nên dễ dàng hơn.

****

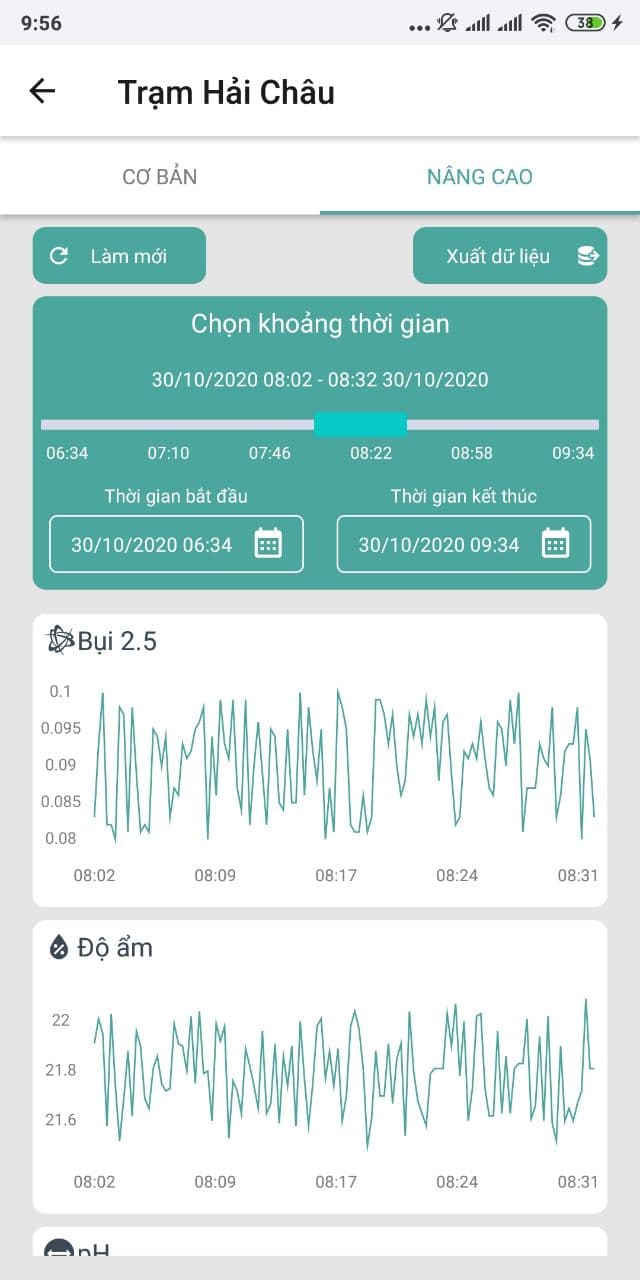
Hình : Ứng dụng công nghệ Blockchain

1. **Ứng dụng web và di động**

Chúng tôi xây dựng một trang web và ứng dụng di động dựa trên dữ liệu được lưu trữ trong Mô-đun 2 và xác thực trong hệ thống blockchain trong Mô-đun 3. Trang web cho phép các tổ chức đăng ký các trạm, tài khoản quản trị và tài khoản người dùng.



Hình : Website quản lý người dùng

****

Hình : Giao diện ứng dụng di động

Ứng dụng di động cung cấp cho người dùng khả năng quản lý danh sách các trạm quan trắc, truy xuất dữ liệu hiện tại và lịch sử.



Hình : Giao diện giám sát và truy xuất dữ liệu quan trắc

# CHƯƠNG 5: KẾT QUẢ VÀ ĐÁNH GIÁ

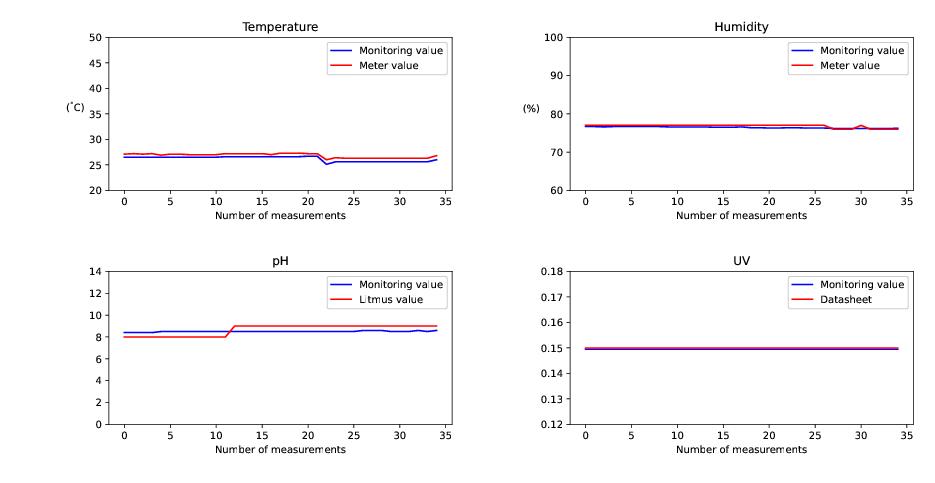
1. **Đánh giá độ chính xác của phần cứng và phần mềm**
   1. **Phần cứng**

Để kiểm tra độ chính xác của các thiết bị, chúng tôi đã chạy thử nghiệm các thiết bị tại Công viên 29 tháng 3, thành phố Đà Nẵng và gửi dữ liệu đến MircoCloud trong vòng hai tuần. Sau đó, chúng tôi so sánh kết quả thu được với các thông số của thiết bị đo nhiệt độ và độ ẩm thu được từ hệ thống đề xuất được so sánh với giá trị của thiết bị đo thông thường. Cường độ tia cực tím và khối lượng bụi trong không khí được so sánh với dữ liệu trong biểu dữ liệu của cảm biến. Nồng độ pH được so sánh với thang đo trên giấy quỳ.

Bảng : Bảng kết quả đo của trạm quan trắc và các thiết bị chuyên dùng

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STT | Độ ẩm | | Nhiệt độ | | UV | | pH | |
| Trạm giám sát | Máy đo | Trạm giám sát | Máy đo | Trạm giám sát | Datasheet (25oC) | Trạm giám sát | Giấy quỳ tím |
| 1 | 76.69 | 77 | 26.5 | 27.1 | 0.15 | 0.15 | 8.4 | 8 |
| 2 | 76.69 | 77 | 26.5 | 27.2 | 0.15 | 0.15 | 8.4 | 8 |
| 3 | 76.59 | 77 | 26.5 | 27.1 | 0.15 | 0.15 | 8.4 | 8 |
| 4 | 76.69 | 77 | 26.5 | 27.2 | 0.15 | 0.15 | 8.4 | 8 |
| 5 | 76.69 | 77 | 26.5 | 26.9 | 0.15 | 0.15 | 8.5 | 8 |
| 6 | 76.69 | 77 | 26.5 | 27.1 | 0.15 | 0.15 | 8.5 | 8 |
| 7 | 76.69 | 77 | 26.5 | 27.1 | 0.15 | 0.15 | 8.5 | 8 |
| 8 | 76.69 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 9 | 76.69 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 10 | 76.69 | 77 | 26.5 | 27.1 | 0.15 | 0.15 | 8.5 | 8 |
| 11 | 76.69 | 77 | 26.5 | 26.8 | 0.15 | 0.15 | 8.5 | 8 |
| 12 | 76.59 | 77 | 26.5 | 26.9 | 0.15 | 0.15 | 8.5 | 8 |
| 13 | 76.59 | 77 | 26.5 | 26.9 | 0.15 | 0.15 | 8.5 | 8 |
| 14 | 76.59 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 15 | 76.59 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 16 | 76.59 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 17 | 76.59 | 77 | 26.5 | 27 | 0.15 | 0.15 | 8.5 | 8 |
| 18 | 76.59 | 77 | 26.6 | 27.2 | 0.15 | 0.15 | 8.5 | 8 |
| 19 | 76.59 | 77 | 26.6 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 20 | 76.59 | 77 | 26.6 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 21 | 76.5 | 77 | 26.6 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 22 | 76.5 | 77 | 26.6 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 23 | 76.5 | 77 | 26.6 | 27 | 0.15 | 0.15 | 8.5 | 9 |
| 24 | 76.59 | 77 | 26.6 | 27.3 | 0.15 | 0.15 | 8.5 | 9 |
| 25 | 76.39 | 77 | 26.6 | 27.3 | 0.15 | 0.15 | 8.5 | 9 |
| 26 | 76.39 | 77 | 26.6 | 27.3 | 0.15 | 0.15 | 8.5 | 9 |
| 27 | 76.29 | 77 | 26.7 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 28 | 76.29 | 77 | 26.7 | 27.2 | 0.15 | 0.15 | 8.5 | 9 |
| 29 | 76.39 | 77 | 25.1 | 26 | 0.15 | 0.15 | 8.5 | 9 |
| 30 | 76.39 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 31 | 76.29 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 32 | 76.29 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 33 | 76.29 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 34 | 76.29 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 35 | 76.29 | 77 | 25.6 | 26.4 | 0.15 | 0.15 | 8.5 | 9 |
| 36 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 37 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 38 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 39 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 40 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 41 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 42 | 76.29 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.6 | 9 |
| 43 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.6 | 9 |
| 44 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.6 | 9 |
| 45 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 46 | 76.19 | 77 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 47 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 48 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.6 | 9 |
| 49 | 76.19 | 76 | 25.6 | 26.3 | 0.15 | 0.15 | 8.5 | 9 |
| 50 | 76.2 | 76 | 26 | 26.8 | 0.15 | 0.15 | 8.6 | 9 |

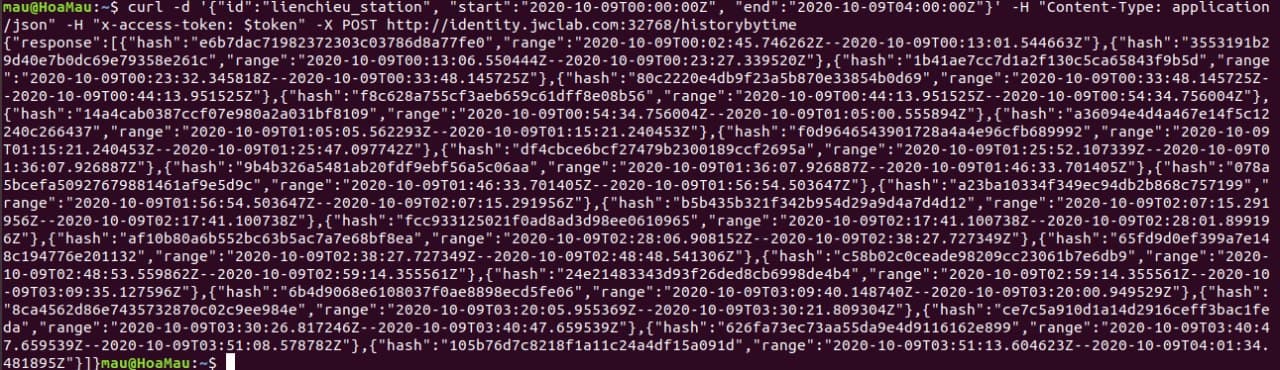
Hình 36 trình bày biểu đồ kết quả của các phép đo. Kết quả cho thấy hệ thống đề xuất hoạt động ổn định, độ chính xác cao.

****

Hình : Đồ thị so sánh kết quả đo

* 1. **Phần mềm**

Hình 37 mô tả dữ liệu được truy vấn từ sổ cái của hệ thống blockchain trong một khoảng thời gian nhất định. Nó đại diện cho một khối dữ liệu được lưu trữ dưới dạng một băm MD5. Dữ liệu thu được từ blockchain có thể là dữ liệu trạng thái thế giới (dữ liệu sổ cái mới nhất), dữ liệu cho một khoảng thời gian cụ thể và tất cả dữ liệu Sổ cái.

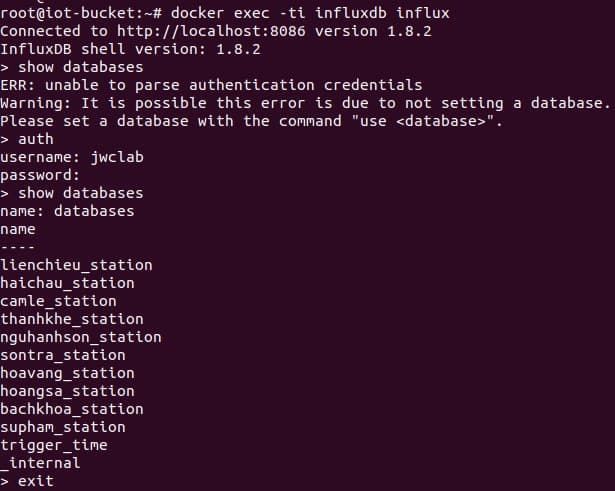
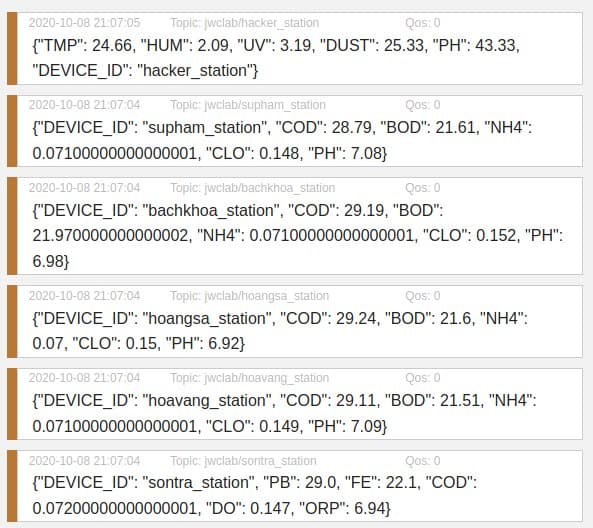
****

Hình : Dữ liệu trong sổ cái của hệ thống Blockchain

1. **Ngăn chặn tấn công từ các trạm quan trắc giả mạo**

Mục đích là để ngăn chặn các trạm giám sát giả mạo xuất bản dữ liệu với định dạng chủ đề không chính xác (ví dụ: định dạng chủ đề chính xác: jwclab / haichau\_station), địa chỉ IP không chính xác và cổng MQTT broker hoặc các trạm chưa đăng ký trong hệ thống blockchain. Một trạm mô phỏng có tên hacker\_station thực hiện một cuộc tấn công giả mạo dữ liệu khi biết IP, cổng và định dạng topic. Hình 38 cho thấy MQTT broker đã nhận được dữ liệu được publish từ trạm hacker\_station, nhưng dữ liệu này không được lưu trữ trong cơ sở dữ liệu chuỗi thời gian vì xác thực từ chuỗi khối hệ thống không được thỏa mãn. Ngược lại, nếu một trạm giả (Mô-đun 1) publish không đúng định dạng topic, sai IP, cổng của broker thì trạm đó không thể tác động đến hệ thống. Có nghĩa là Mô-đun 1 không thể tạo bất kỳ kết nối nào với Mô-đun 2.

Hình 38 cho thấy danh sách cơ sở dữ liệu của các trạm đã đăng ký với hệ thống blockchain. Hacker\_station chưa được xác thực bởi hệ thống blockchain và không thể tạo bất kỳ cơ sở dữ liệu nào.

****

Hình : Trạm quan trắc giả mạo

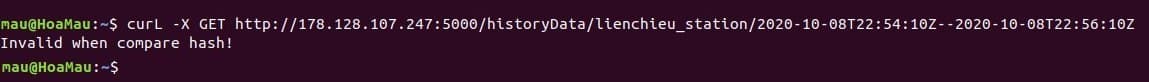
Ngoài các yêu cầu xác thực chính ở trên, chúng tôi cũng xây dựng các hình thức xác thực khác với mức độ bảo mật cao hơn:

* Xác thực với tên trạm (tên trạm publish phải khớp với tên trạm trong danh sách của sổ cái).
* Xác thực dựa trên danh tính của quản trị viên trạm (Khi một tổ chức tham gia vào hệ thống blockchain, họ có quyền cấp chứng chỉ phi tập trung cho người dùng hoặc quản trị viên trạm)

Khi mỗi trạm publish dữ liệu, nó phải trải qua các bước xác thực và lưu dấu vết trong hệ thống blockchain. Nó làm tối ưu thời gian xử lý các dịch vụ và nguy cơ publish dữ liệu không hợp lệ từ các trạm giám sát giả mạo.

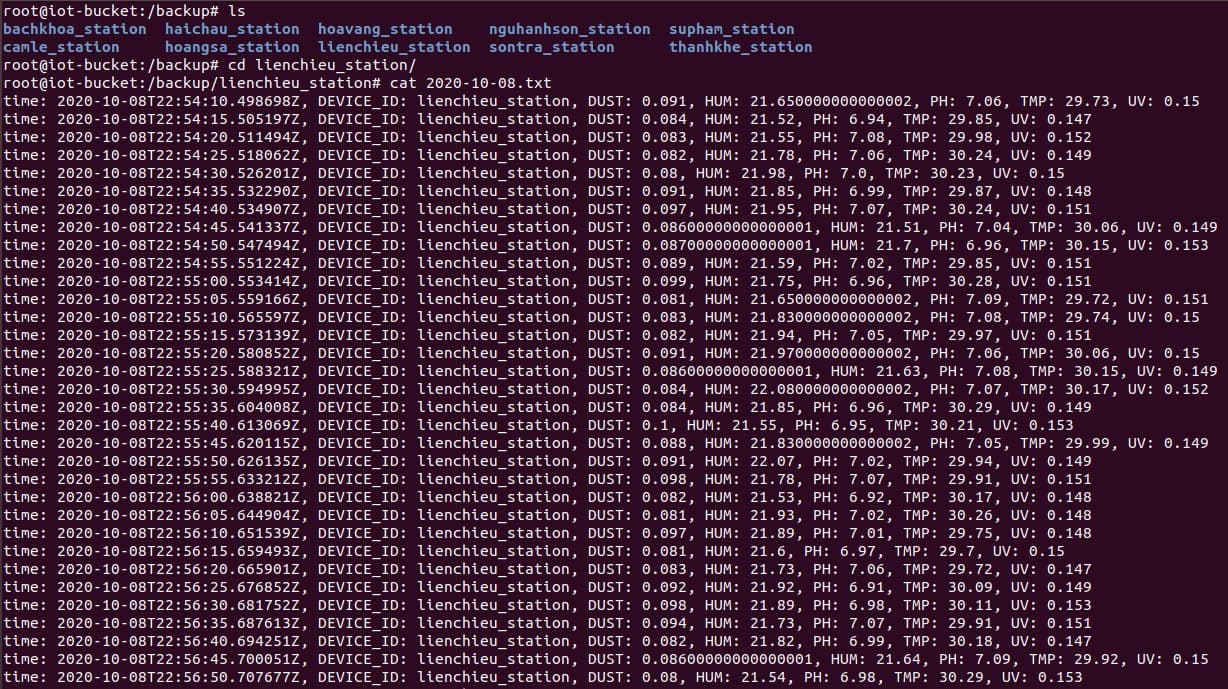
1. **Ngăn chặn phá hủy dữ liệu trên time-series database**

Trong thử nghiệm này, giả sử người tấn công có quyền truy cập vào cơ sở dữ liệu của lienchieu\_station (ví dụ: kẻ tấn công đã đánh cắp tài khoản quản trị và thực hiện các bước xác thực). Kẻ tấn công sẽ thực hiện các thay đổi trái phép đối với dữ liệu trong cơ sở dữ liệu chuỗi thời gian đó. Tuy nhiên, khi người dùng truy vấn dữ liệu vừa được thay đổi, các microservices sẽ so sánh hàm băm của dữ liệu mới với dữ liệu đã được lưu trữ và xác thực trước đó trong hệ thống blockchain. Hệ thống sẽ đưa ra cảnh báo về sự khác biệt giữa các mã băm của khối dữ liệu trong cơ sở dữ liệu (Hình 39).



Hình : Xác thực dữ liệu trong Blockchain và InfluxDB

Chúng ta tiến hành khôi phục dữ liệu gốc bằng back up service (Hình 40). Cần lưu ý rằng, mặc dù hệ thống được đề xuất có thể phát hiện những thay đổi trái phép và có khả năng thực hiện khôi phục cơ sở dữ liệu, nhưng khả năng rủi ro này xảy ra là rất thấp. Để đạt được sự thay đổi dữ liệu này, những kẻ tấn công cần chiếm quyền quản trị tại MicroCloud và thông qua tắt cả các bước xác thực.



Hình : Dữ liệu được sao lưu về máy từ cơ sở dữ liệu

# CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. **Kết luận**

Trong bài báo này, chúng tôi đề xuất kết hợp IoT và công nghệ blockchain để thực hiện một hệ thống giám sát môi trường đáng tin cậy. Dữ liệu giám sát được thu thập bởi các cảm biến và được lưu trữ trong cơ sở dữ liệu chuỗi thời gian với sự xác thực của hệ thống blockchain. Nó sẽ đảm bảo dữ liệu quan trắc môi trường được an toàn và minh bạch. Chúng tôi đã thực hiện nhiều thử nghiệm khác nhau để xác minh hoạt động của hệ thống được đề xuất:

- Xác minh độ chính xác trong các phép đo

- Thực hiện chống tấn công từ các trạm giám sát giả mạo

- Ngăn chặn tấn công thay đổi dữ liệu trong cơ sở dữ liệu chuỗi thời gian

1. **Hướng phát triển**

Trong tương lai, chúng tôi sẽ triển khai các cảm biến khác trong các môi trường cụ thể cho các mục đích khác nhau. Bên cạnh đó, chúng tôi hy vọng sẽ cải thiện các khía cạnh bảo mật của công nghệ này để giảm nguy cơ tấn công mạng và xâm nhập dữ liệu.

# TÀI LIỆU THAM KHẢO

1. Cliﬀord Ho, Alex Robinson, David Miller, and Mary Davis. 2005. Overview of Sensors and Needs for Environmental Monitoring. Sensors 5, 1 (Feb. 2005), 4–37. https://doi.org/10.3390/s5010004
2. Luís M. Oliveira and Joel J. Rodrigues. 2011. Wireless Sensor Networks: a Survey on Environmental Monitoring.  
   Journal of Communications 6, 2 (April 2011). https://doi.org/10.4304/jcm.6.2.143-151
3. https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf
4. https://mlab.vn/index.php?\_route\_=47089-huong-dan-su-dung-module-sim800a-phan-2-giao-tiep-voi-server.html
5. https://cdn.sparkfun.com/datasheets/Sensors/LightImaging/ML8511\_3-8-13.pdf
6. https://mechasolution.vn/Blog/bai-27-cam-bien-bui-pm2-5-dust-sensor-gp2y10
7. Meena Singh, M.A. Rajan, V.L. Shivraj, and P. Balamuralidhar. 2015. Secure MQTT for Internet of Things (IoT). In 2015Fifth International Conference on Communication Systems and Network Technologies. IEEE. https://doi.org/10.1109/csnt. 2015.16
8. Axel Moinet, BenoÃőt Darties, and Jean-Luc Baril. 2017. Blockchain based trust and authentication for decentralized  
   sensor networks. arXiv:1706.01730 [cs.CR]
9. Jingyu Zhang, Siqi Zhong, Tian Wang, Han-Chieh Chao, and Jin Wang. 2020. Blockchain-based systems and applications:  
   A survey. Journal of Internet Technology 21, 1 (2020), 1–14.
10. Marco Conoscenti, Antonio Vetro, and Juan Carlos De Martin. 2016. Blockchain for the Internet of Things: A systematic literature review. In 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA). IEEE.https://doi.org/10.1109/aiccsa.2016.7945805
11. Syeda Noor Zehra Naqvi, Sofa Yfantidou, and Esteban Zimányi. 2017. Time series databases and inﬂuxdb. (2017).
12. https://obrienlabs.net/how-to-setup-your-own-mqtt-broker/
13. https://stackoverflow.com/questions/45952125/python-push-notification-through-fcm-apns-to-ios-or-android-devices
14. https://hyperledger-fabric.readthedocs.io/en/release2.0/smartcontract/ smartcontract.html
15. Omar Al-Debagy and Peter Martinek. 2018. A Comparative Review of Microservices and Monolithic Architectures. In 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI). IEEE. https://doi.org/10.1109/cinti.2018.8928192
16. https://medium.com/@arunrajeevan/transaction-flow-in-hyperledger-b7f9ee43ad60
17. https://viblo.asia/p/bai-3-cac-khai-niem-co-ban-trong-hyperledger-fabric-gDVK2p3mlLj
18. Jingyu Zhang, Siqi Zhong, Tian Wang, Han-Chieh Chao, and Jin Wang. 2020. Blockchain-based systems and applications: A survey. Journal of Internet Technology 21, 1 (2020), 1–14.
19. Gbadebo Ayoade, Vishal Karande, Latifur Khan, and Kevin Hamlen. 2018. Decentralized IoT Data Management Using BlockChain and Trusted Execution Environment. In 2018 IEEE International Conference on Information Reuse andIntegration (IRI). IEEE. https://doi.org/10.1109/iri.2018.00011