**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**BÁO CÁO PROJECT III**

**Ứng dụng Blockchain trong IOT**

**đảm bảo tính xác thực thông tin**

**NGUYỄN TUẤN HIỆP**

hiep.nt170070@sis.hust.edu.vn

**Chương trình tài năng - Công nghệ thông tin K62**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | TS. Trần Hải Anh |
| **Bộ môn:**  **Viện:** | Truyền thông và mạng máy tính  Công nghệ thông tin |

# Chương I : Lí do chọn đề tài

Trong thời đại công nghệ bùng nổ, Internet of Things (IOT) cũng phát triển mạnh mẽ, len lỏi vào trong tất cả các ngành công nghiệp, đời sống, kết nối mọi thứ với nhau. IOT phát triển trên một hệ thống internet, người dùng tương tác với các thiết bị thông qua mạng. Cùng với sự phát triển đó, một vấn đề cấp thiết đặt ra là phải đảm báo tính an toàn cho hệ thống IOT.

Blockchain là hệ thống cơ sở dữ liệu lưu trữ thông tin theo dạng khối dựa trên nền tảng mã hóa. Blockchain có tính bảo mật cao, đảm bảo thông tin không bị sửa đổi dưới hình thức xác thực phân tán. Công nghệ này có thể áp dụng vào IOT để đảm bảo tính xác thực thông tin trong mạng lưới, nâng cao tính bảo mật của hệ thống.

Trong đề án này, tôi sẽ trình bày cách thức áp dụng công nghệ blockchain trong IOT, so sánh hiệu năng với mạng IOT thông thường. Đồng thời, xây dựng website minh họa cho hệ thống.

Nội dung

[Chương I : Lí do chọn đề tài 2](#_Toc60188805)

[Chương II : Cơ sở lí thuyết 4](#_Toc60188806)

[II.1 Internet of Things 4](#_Toc60188807)

[II.1.1 Tổng quan về IOT 4](#_Toc60188808)

[II.1.2 Truyền thông giữa các thiết bị 5](#_Toc60188809)

[II.2 Blockchain 6](#_Toc60188810)

[II.3 Blockchain trong IOT 7](#_Toc60188811)

[Chương III : Phân tích thiết kế hệ thống 9](#_Toc60188812)

[III.1 Sơ đồ thiết kế tổng quan 9](#_Toc60188813)

[III.2 Sơ đồ usecase phân rã 10](#_Toc60188814)

[III.3 Danh sách usecase 11](#_Toc60188815)

[III.3.1 Sơ đồ trình tự usecase UC04 11](#_Toc60188816)

[III.3.2 Sơ đồ trình tự usecase UC05 12](#_Toc60188817)

[III.3.3 Sơ đồ trình tự usecase UC07 12](#_Toc60188818)

[III.3.4 Sơ đồ trình tự usecase UC08 13](#_Toc60188819)

[Chương IV : Triển khai thực hiện 14](#_Toc60188820)

[IV.1 Things 14](#_Toc60188821)

[IV.2 Blockchain 15](#_Toc60188822)

[IV.3 Web of Things 17](#_Toc60188823)

[IV.4 Xác thực bằng blockchain 19](#_Toc60188824)

[Chương V : Kết quả thực nghiệm 20](#_Toc60188825)

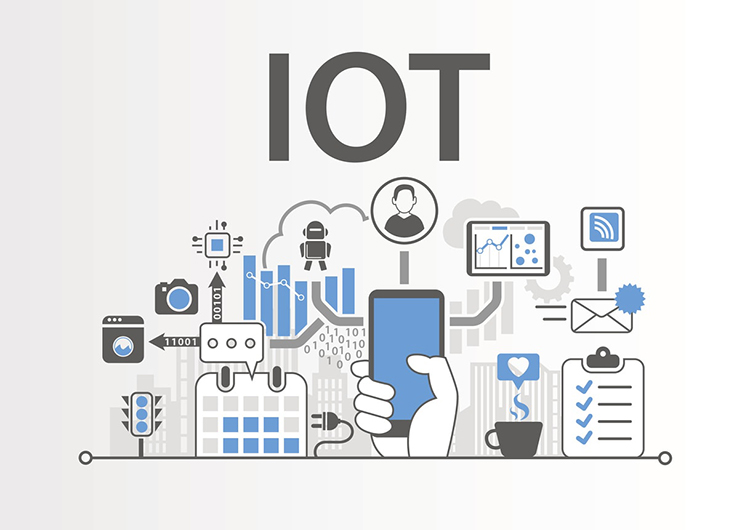
[Chương VI : Kết luận 21](#_Toc60188826)

# Chương II : Cơ sở lí thuyết

## II.1 Internet of Things

### II.1.1 Tổng quan về IOT

Internet of things (IOT), internet vạn vật là một mạng lưới thông tin kết nối các thiết bị vật lý với nhau, thu thập và cùng chia sẻ dữ liệu. Nhờ bộ xử lý giá rẻ, mạng internet mà mọi thiết bị đều có khả năng kết nối vào mạng lưới. Điều này dẫn tới một cuộc cách mạng về chuyển đổi số, đem lại những tiện ích vô cùng lớn cho người dùng khi mọi thao tác đều có thể kiểm soát qua mạng internet, mọi lúc, mọi nơi.

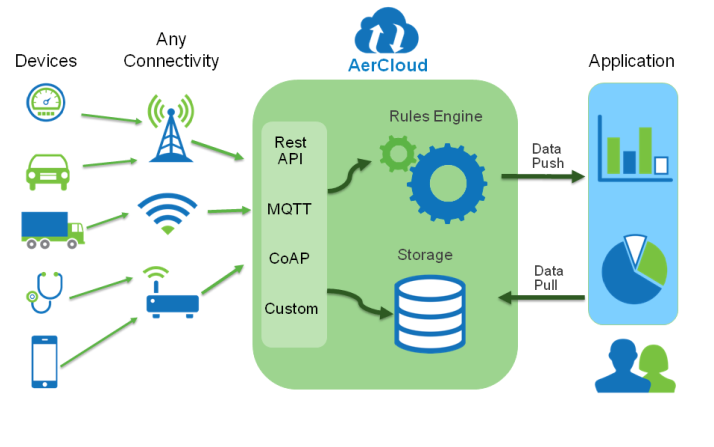


Hình II.1 Ví dụ về IOT

Một ví dụ nhỏ về ứng dụng của IOT trong nông nghiệp, người ta có thể xây dựng một hệ thống nông nghiệp thông minh. Trong đó, các thiết bị sẽ tiến hành đo đạc các thông số về môi trường như nhiệt độ, độ ẩm, độ màu mỡ, … Sau đó, gửi lên máy chủ để xử lý thông tin, từ đó, đưa ra các mệnh lệnh tới các máy để thực hiện các tác vụ như tăng giảm nhiệt độ, tưới nước, …. Với nông nghiệp thông minh, người nông dân sẽ không cần trực tiếp ở trên nông trường, làm các công việc thủ công. Do đó, tiết kiệm thời gian, tăng hiệu suất làm việc, …

### II.1.2 Truyền thông giữa các thiết bị

Đặc điểm chính của các thiết bị IOT là việc tương tác, truyền thông với mạng internet. Dữ liệu từ các thiết bị vào cần được server xử lí và gửi đến các thiết bị ra. Do các thiết bị cần liên tục cập nhật dữ liệu, truyền chúng qua mạng rồi lại gửi lệnh nên các chuẩn truyền thông trên IOT cũng khác.

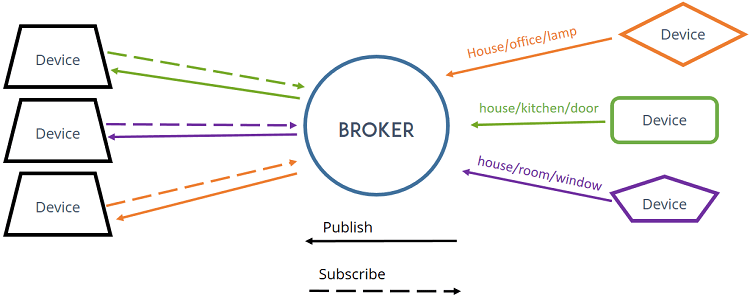


Hình II.2 Truyền thông giữa các thiết bị IOT

Các chuẩn truyền thông trên IOT phổ biến nhất là :

* MQTT (Message Queue Telemetry Transport)
* CoAP (Constrained Applications Protocol)
* AMQP (Advanced Message Queue Protocol)
* XMPP (Extensible Messaging và Presence Protocol)
* DDS (Data Distribution Service)

MQTT (Message Queue Telemetry Transport) là một giao thức mã nguồn mở nhằm truyền tải thông tin giữa các máy khách thông qua một broker trung gian. MQTT hoạt động theo nguyên tắc Publisher/Subsriber. MQTT là một giao thức khá nhẹ, hoạt động đơn giản, dễ sử dụng, tương thích với các thiết bị tiêu thụ điện năng thấp.



Hình II.3 MQTT

Trong đề tài này, tôi sử dụng giao thức MQTT để thực hiện việc giao tiếp giữa các thiết bị.

## II.2 Blockchain

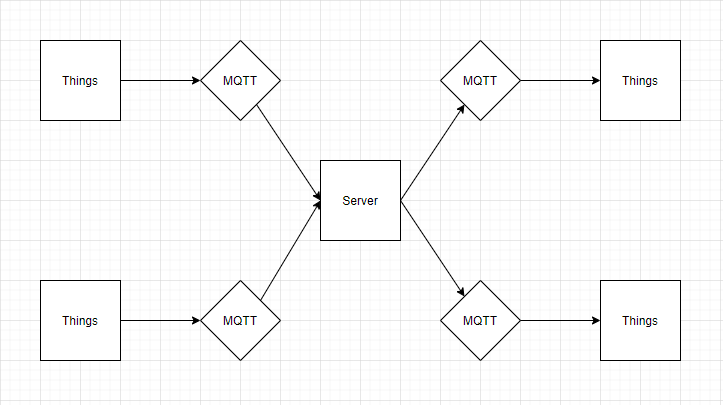
Blockchain là hệ thống phân tán, các nút đều có quyền truy cập cơ sở dữ liệu, trao đổi trực tiếp với nhau và không thông qua bên thứ ba. Một khi giao dịch được thực hiện, thông tin giao dịch được lưu vào cơ sở dữ liệu thì sẽ được đảm bảo là không thể bị sửa đổi. Từ đó có thể chứng thực được thông tin.

Một người khi muốn tham gia vào mạng lưới bitcoin sẽ được cấp phát 1 cặp khóa public - private. Khi muốn thực hiện một giao dịch, đơn giản người dùng sẽ kí bằng khóa bí mật. Tiếp đến, mạng lưới sẽ xác nhận bằng khóa công khai rồi thêm vào cơ sở dữ liệu. Khi giao dịch đã được thực hiện thành công, không có cách nào thay đổi được thông tin về giao dịch.

Ethereum là một nền tảng điện toán có tính chất phân tán dựa trên công nghệ Blockchain.

## II.3 Blockchain trong IOT

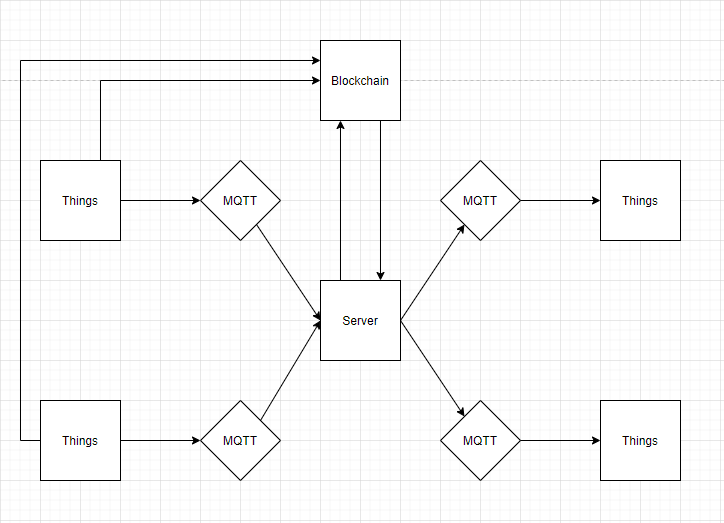
Thiết kế hệ thống IOT truyền thông bằng giao thức MQTT thông thường :



Hình II.4 IOT với MQTT

Các thiết bị đơn giản chỉ cần truyền thông tin tới server thông qua giao thức MQTT. Sau đó, server tiến hành xử lí rồi lại tiếp tục gửi lệnh tới các thiết bị things. Giao thức này chưa có các cơ chế bảo mật, xác thực cũng như tính toàn vẹn thông tin. Giả sử như có kẻ tấn công vào giao thức MQTT làm thay đổi thông tin luồng dữ liệu gửi tới server dẫn tới việc xử lí sai, nhiều khi sẽ gây ra hậu quả nghiêm trọng.

Thiết kế hệ thống IOT sử dụng công nghệ blockchain :



Hình II.5 IOT với blockchain

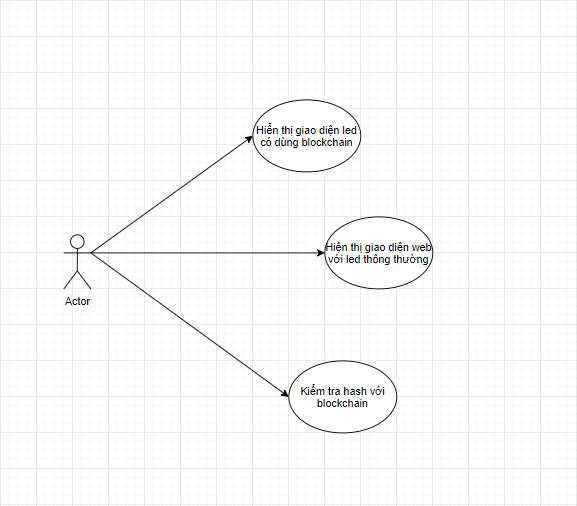
Những thông tin, giao dịch quan trọng sẽ được lưu trữ trong blockchain. Server sẽ thực hiện việc xác thực người dùng theo hai cách :

* Đối với những thao tác quan trọng, các thiết bị sẽ đẩy mọi thông tin lên blockchain, server trước khi thực hiện sẽ xác thực thông tin với mạng blockchain
* Đối với nhưng thông tin không quan trọng, các thiết bị vẫn sẽ tiến hành đẩy thông tin lên blockchain. Server sẽ thực hiện tính toán mà không cần xác thực. Sau đó, nếu cần xác thực lại sẽ xác nhận với mạng blockchain.

Cách một thời gian tiến hành sẽ lâu hơn, nên những thao tác yêu cầu độ chính xác cao sẽ dùng đến cách này.

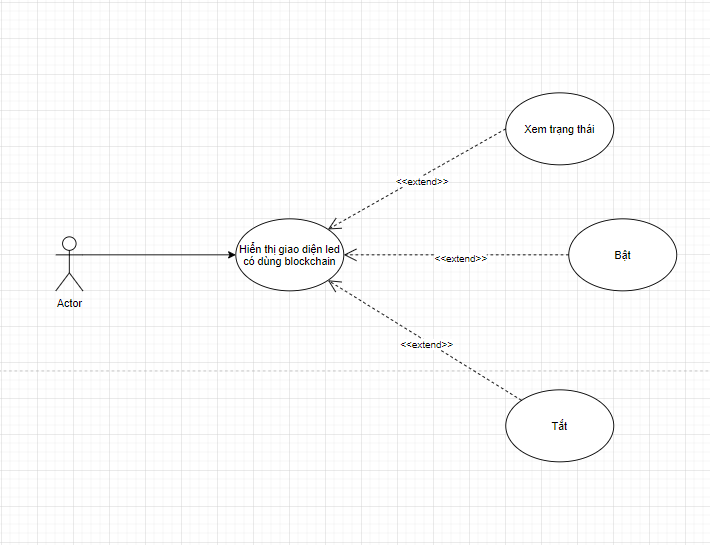
# Chương III : Phân tích thiết kế hệ thống

## III.1 Sơ đồ thiết kế tổng quan

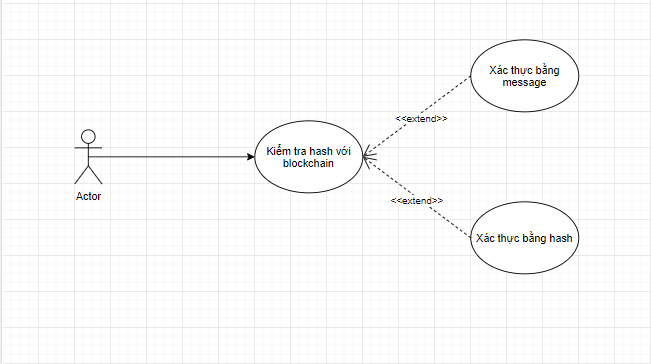


Hình III 1 Sơ đồ usecase tổng quan

## III.2 Sơ đồ usecase phân rã



Hình III 2 Sơ đồ usecase phân rã "Hiển thị giao diện led có dùng blockchain"

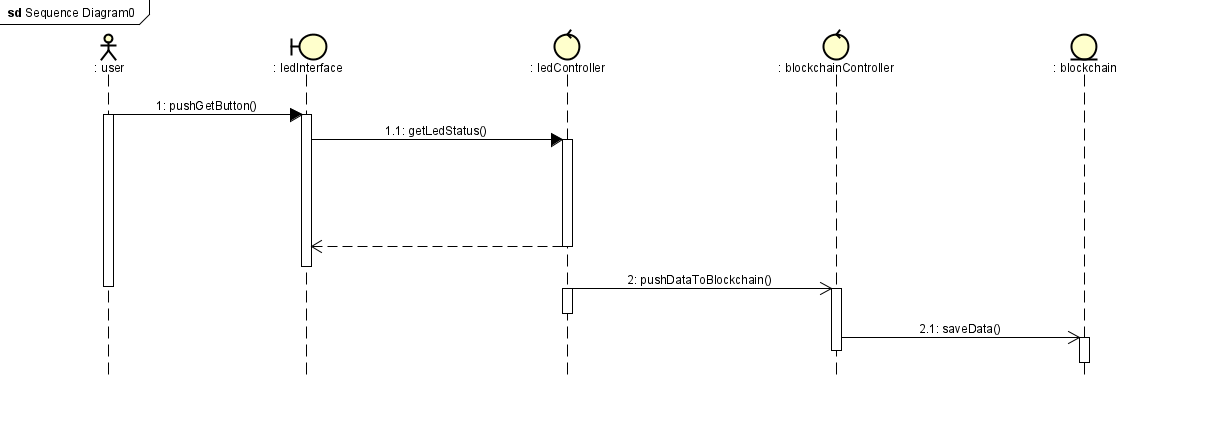


Hình III 3 Sơ đồ usecase phân rã "Kiểm tra hash với blockchain"

## III.3 Danh sách usecase

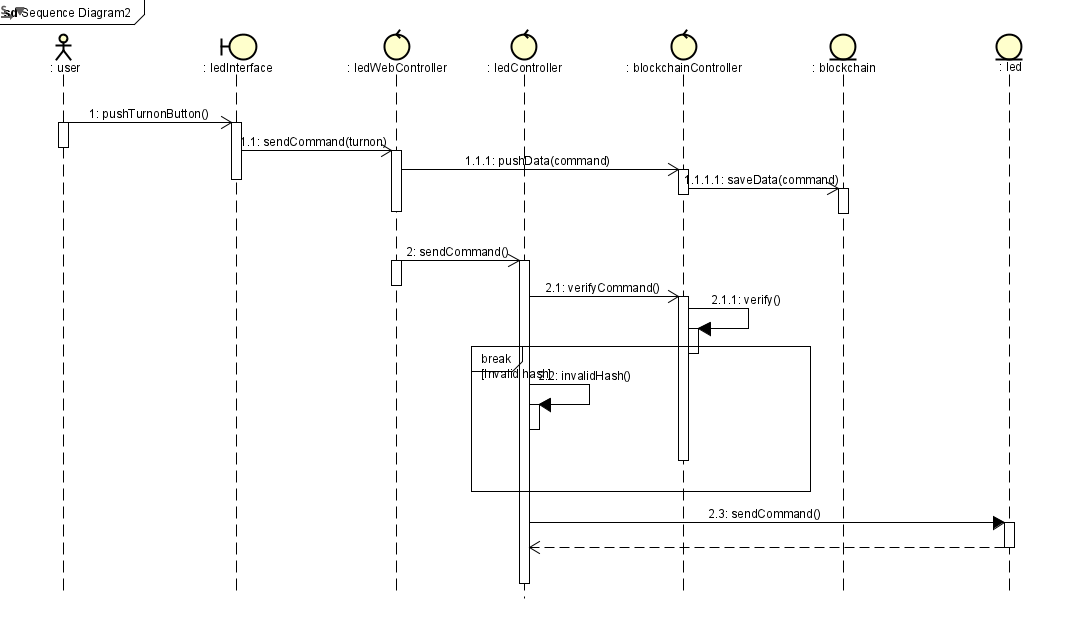
|  |  |  |
| --- | --- | --- |
| **Mã usecase** | **Tên usecase** | **Chức năng** |
| UC01 | Hiển thị giao diện led có dùng blockchain | Hiển thị giao diện web có dùng blockchain |
| UC02 | Hiển thị giao diện led không dùng blockchain | Hiển thị giao diện web không dùng blockchain |
| UC03 | Kiểm tra hash với blockchain | Hiển thị giao diện xác thực với hệ thống blockchain bằng mã hash |
| UC04 | Xem trạng thái LED | Xem trạng thái bật tắt của led |
| UC05 | Bật LED | Thực hiện gửi các gói tin yêu cầu bật led |
| UC06 | Tắt LED | Thực hiện gửi các gói tin yêu cầu tắt led |
| UC07 | Xác thực bằng message | Cho phép người dùng xác thực với hệ thống blockchain một đoạn message có tồn tại |
| UC08 | Xác thực bằng hash | Xác thực một mã hash có tồn tại trong hệ thống |

### III.3.1 Sơ đồ trình tự usecase UC04



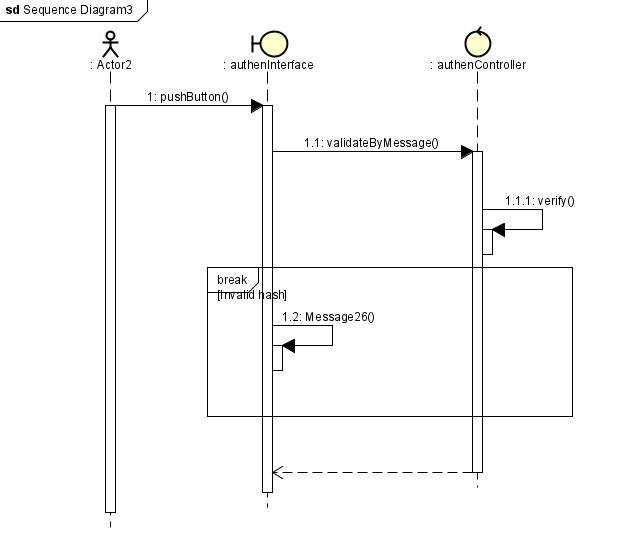
Hình III 4 Sơ đồ trình tự usecase UC04

### III.3.2 Sơ đồ trình tự usecase UC05



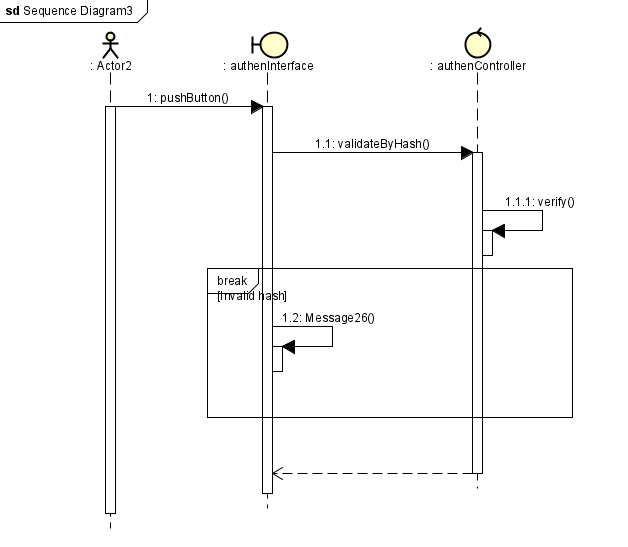
Hình III 5 Sơ đồ trình tự usecase UC05

### III.3.3 Sơ đồ trình tự usecase UC07



Hình III 6 Sơ đồ trình tự usecase UC07

### III.3.4 Sơ đồ trình tự usecase UC08



Hình III 7 Sơ đồ trình tự usecase UC08

# Chương IV : Triển khai thực hiện

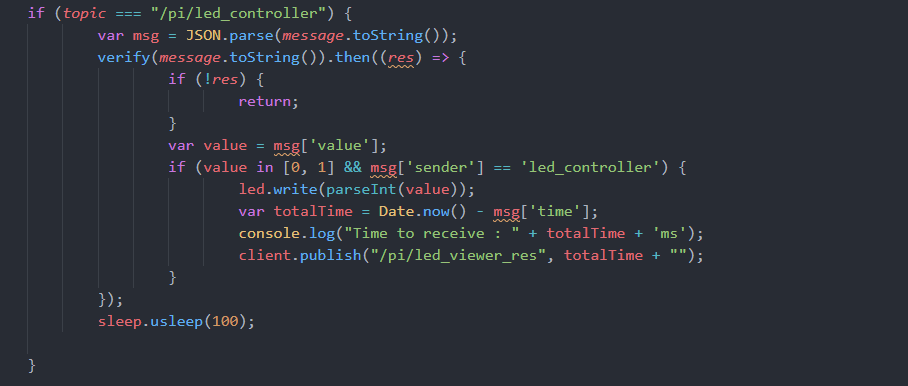
## IV.1 Things

Ngôn ngữ lập trình tôi sử dụng ở đây là javascript với framework nodejs. Nodejs giúp tôi dễ dàng kết nối với các thiết bị IOT, mqtt, mạng lưới blockchain cũng như tạo ra một website đơn giản nhờ vào hệ thống thư viện rất đồ sộ.

Để điều khiển led, tôi sử dụng thư viện **onoff**. Tiến hành kết nối led với cồng Gpio 26 của rasberry Pi.

Tạo server MQTT bằng **mosquito** chạy trên port 1028. Sau đó dùng thư viện **mqtt** của **nodejs** để tiến hành public và subcribe các message.

Chương trình điều khiển led sẽ tiến hành nghe các câu lệnh điều khiển nhận từ server, phân tích và đưa ra các hành động tương ứng với mệnh lệnh đề ra.



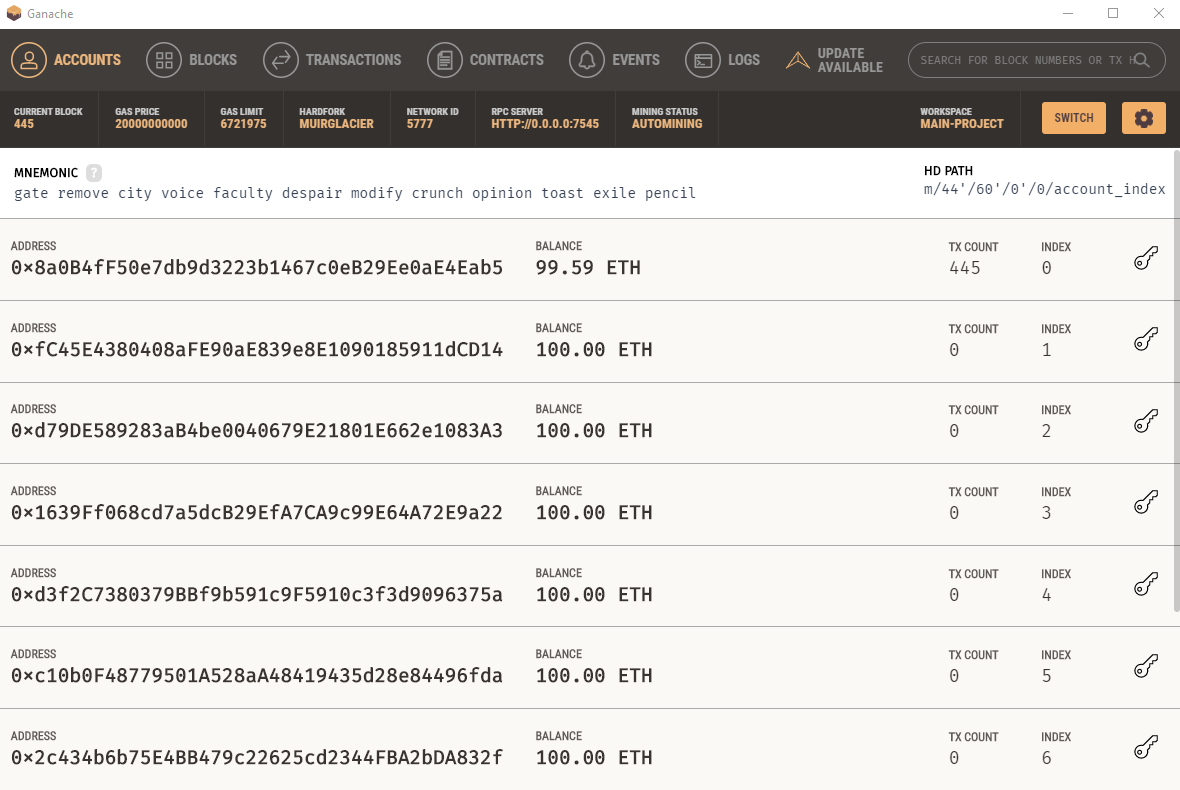
Hình IV 1 bật tắt đèn có validate bằng blockchain

Chương trình sẽ thực hiện lệnh khi nhận được message từ topic **/pi/led\_controller**. Nó sẽ tiến hành phân tích message được gửi tới, chuyển nó về dạng JSON, tiến hành các thao tác validation bằng hệ thống blockchain. Nếu hệ thống trả về là hash hợp lệ, chương trình sẽ tiếp tục một số validation về giá trị của tham số cũng như người gửi, sau đó mới tiến hành bật tắt led. Cuối cùng gửi trả lại thông tin về thời gian hoàn thành thông qua MQTT.

## IV.2 Blockchain

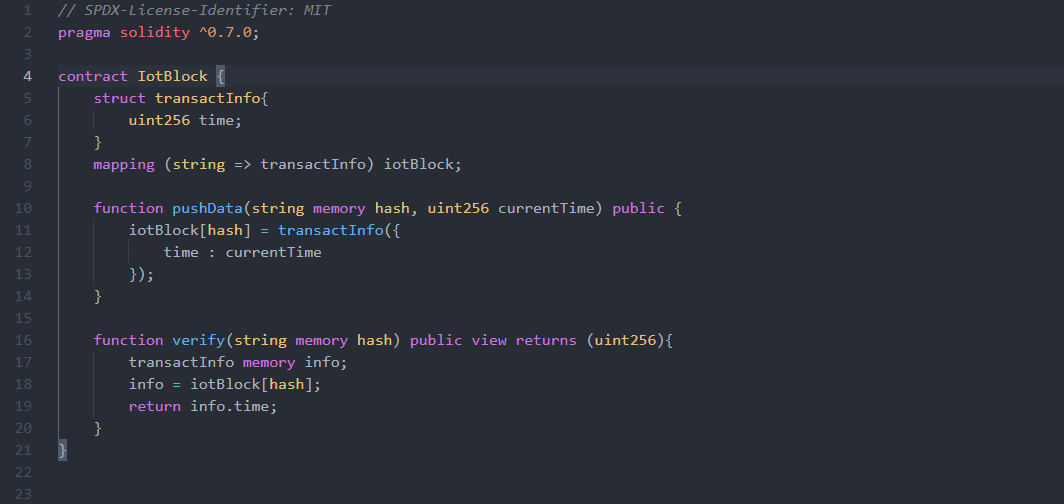
Như đã trình bày ở trên, trong bài này tôi sử dụng hệ thống tiền ảo ethereum với sự giúp đỡ của ganache và solidity.

Ganache sẽ tạo 10 account với mỗi account có sẵn 100 ETH. Đồng thời bằng giao diện, chúng ta cũng có thể xem lại các giao dịch đã thực hiện. Đồng thời, có thể setup thành mạng local để các thiết bị trong cùng mạng có thể truy cập được. Do đó, hệ thống này rất tiện lợi để bắt đầu mô phỏng với blockchain.



Hình IV 2 Ganache mô phỏng mạng lưới blockchain

Để có thể sử dụng blockchain để lưu trữ thông tin các giao dịch trong hệ thống IOT, chúng ta cần sử dụng các smart contract. Tại đây tôi đã sử dụng solidity 0.7.0 để tạo một smart contract đơn giản **IotBlock**.



Hình IV 3 Solidity smartcontract

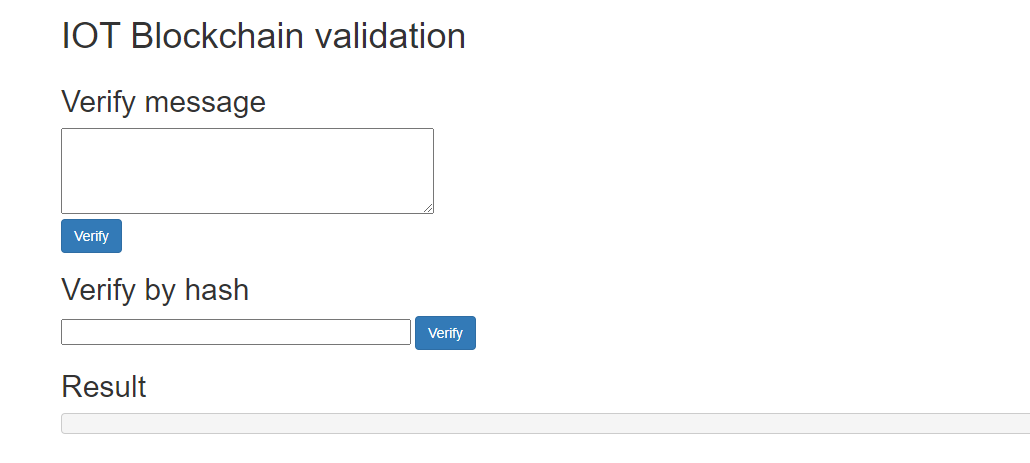
Smartcontract này gồm 2 hàm chính là **pushData** và **verify**.

* Hàm **pushData** cho phép chúng ta lưu giữ một hash vào trong mapping của solidity. Mapping này sẽ được map với tham số thứ hai : currentTime. **CurrentTime** là thời gian tạo transaction.
* Hàm **verify** cho phép trả về thời gian tương ứng với từng hash, nếu hash tồn tại. Do đó, khi verify một message tôi sẽ tiến hành mã hóa đoạn message đó, check xem hash có tồn tại trong mapping hay không, rồi so sánh giá trị thời gian mà hàm verify trả về. Điều này vừa đảm bảo được tính toàn vẹn của message đồng thời tránh bị tấn công dùng lại, đảm bảo tính chính xác về thời gian.

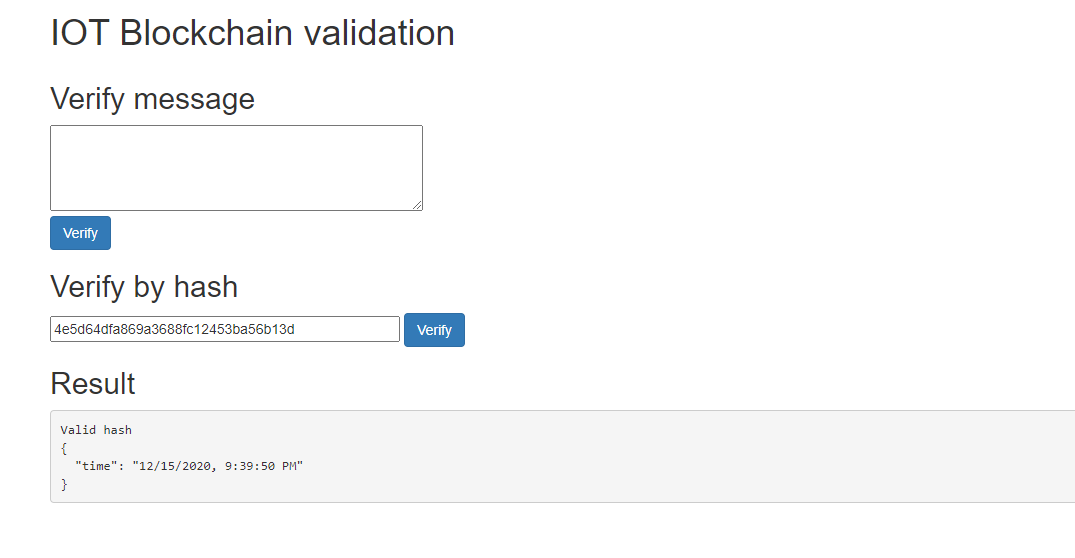
## IV.3 Web of Things

Tôi có xây dựng một hệ thống Web để hiển thị giao diện cho hệ thống này.

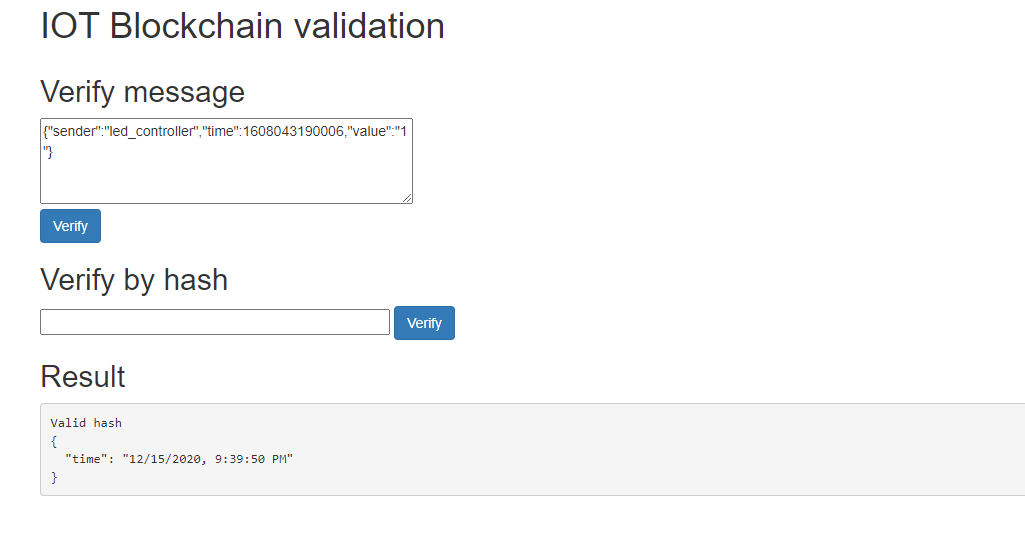
Tôi có một trang để nhập mã hash, tiến hành các thao tác validation với các message bằng blockchain.



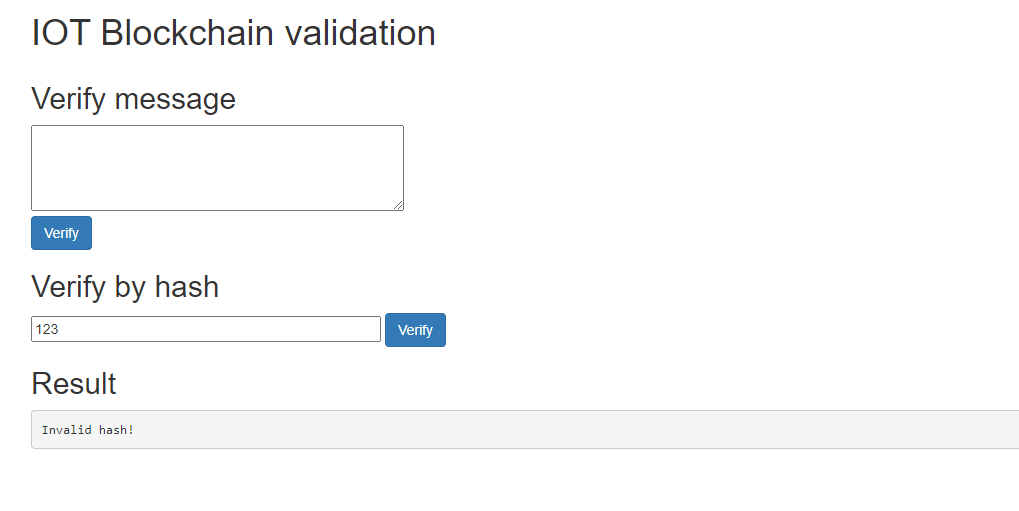
Hình IV 4 Web interface message validation



Hình IV 5Message validation by hash

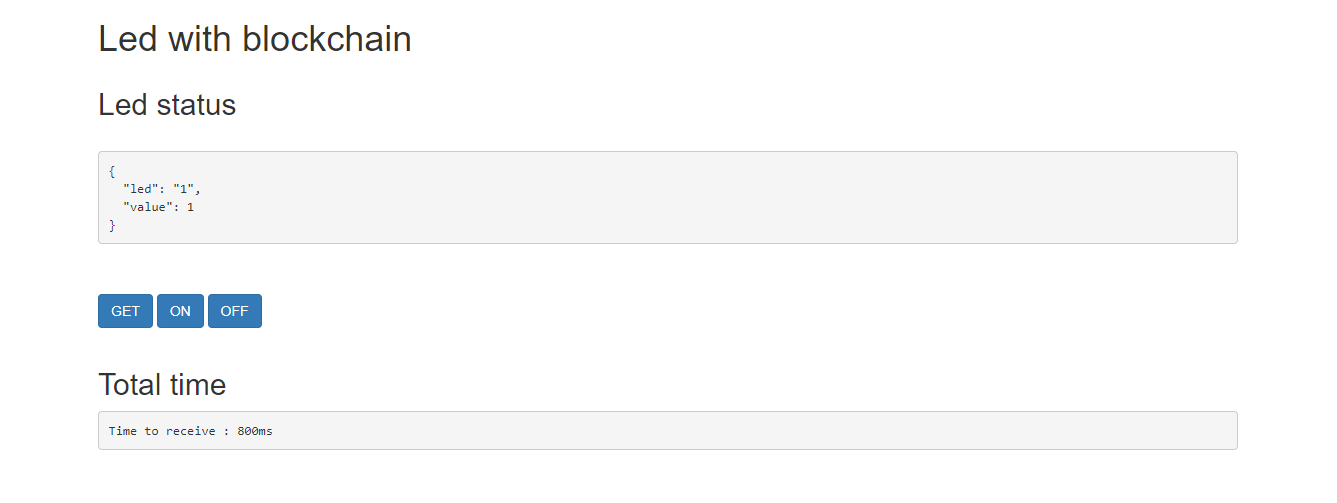


Hình IV 6 Message validation by json



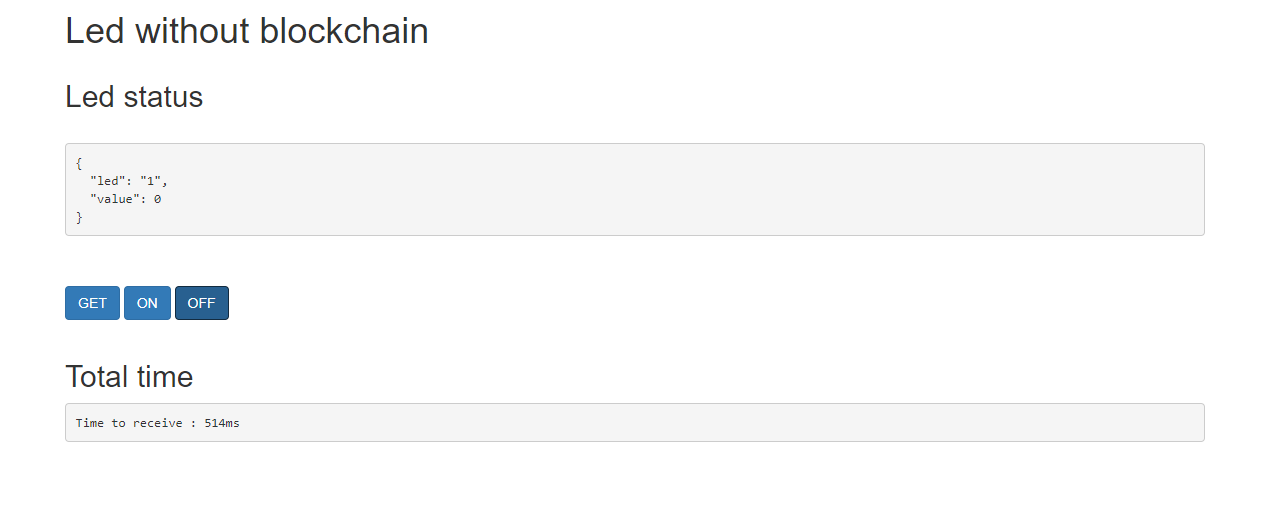
Hình IV 7Unauthen message

Tôi cũng xây dựng một giao diện website thực hiện hiển thị thông tin trạng thái bật tắt của led có sử dụng blockchain. Chúng ta có thể tiến hành bật tắt ngay trên trang này. Đồng thời, nó cũng hiển thị thời gian thực hiện của thao tác này.



Hình IV 8 Led controller with blockchain

Tôi cũng xây dựng một giao diện website để thực hiện hiển thị led không sử dụng blockchain để so sánh.



Hình IV 9 Led controller without blockchain

## IV.4 Xác thực bằng blockchain

Tôi chia ra thành 2 loại xác thực tùy vào từng tác động của hành động :

* Đối với những hành động không quan trọng : là những thông tin về dữ liệu được cập nhật thường xuyên, liên tục Do mang tính chất dữ liệu nên không cần chính xác ngay tại thời điểm tới. Do đó, tôi vẫn sẽ thực hiện truyền và nhận toàn bộ thông tin. Nhưng điểm khác biệt là tôi sẽ đẩy hash của dữ liệu vào trong blockchain. Đến một thời điểm trong tương lai, tôi cần xác thực message thì sẽ dựa vào hash của message cùng với smartcontract đã viết ở bên trên để xác thực
* Đối với những hành động quan trọng : Là những hành động cần xác thực ngay trước khi thực hiện, trong ví dụ này là bật/ tắt đèn. Do đó, trước khi truyền đạt yêu cầu, người gửi sẽ đẩy hash của message vào trong blockchain, sau đó khi bên nhận nhận được message sẽ tiến hành xác nhận với hệ thống. Cuối cùng, sau khi được chứng thực thì mới thực hiện yêu cầu đưa ra.

# Chương V : Kết quả thực nghiệm

Việc truyền nhận dữ liệu led sử dụng giao thức xác thực 1 sẽ không làm thay đổi thời gian truyền, nhận data do việc đẩy vào blockchain được thực hiện sau.

Bật tắt led :

|  |  |  |
| --- | --- | --- |
| **STT** | **Led có blockchain** | **Led không có blockchain** |
| 1 | 1172 ms | 512 ms |
| 2 | 898 ms | 350 ms |
| 3 | 846 ms | 362 ms |
| 4 | 802 ms | 368 ms |
| 5 | 862 ms | 358 ms |

Ta thấy việc thêm blockchain vào sẽ làm tăng thời gian xử lí nhưng không quá nhiều.

# Chương VI : Kết luận

Hệ thống Blockchain khi được ứng dụng vào IOT sẽ làm tăng tính xác thực của hệ thống lên nhiều. Đồng thời, nó cũng không làm tăng quá nhiều thời gian trễ. Tuy nhiên, chúng ta cũng cần chọn lọc, áp dụng vào từng trước hợp cụ thể để đạt được hiệu quả tối ưu. Một nhược điểm của hệ thống này là chi phí sẽ tăng cao do phải duy trì hệ thống blockchain, lưu trữ dữ liệu hash. Đồng thời, số lượng message quá lớn của hệ thống IOT sinh ra mỗi ngày sẽ khó có thể duy trì lâu dài. Blockchain cho IOT vẫn đang ở giai đoạn nghiên cứu, thử nghiệm chưa được ứng dụng rộng rãi nhưng với những tiềm năng của nó, ta vẫn có thể hi vọng nó sẽ được ứng dụng trong một tương lai không xa.