

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ SÀI GÒN  
KHOA CÔNG NGHỆ THÔNG TIN  
---oOo---

**LUẬN VĂN TỐT NGHIỆP**

*Tên đề tài:*

**TÌM HIỂU CÔNG NGHỆ  
BLOCKCHAIN**

TP HỒ CHÍ MINH – NĂM 2018

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ SÀI GÒN  
KHOA CÔNG NGHỆ THÔNG TIN  
---oOo---

## LUẬN VĂN TỐT NGHIỆP

Tên đề tài:

# TÌM HIỂU CÔNG NGHỆ BLOCKCHAIN

Người hướng dẫn : NGÔ THỊ BẢO TRÂN  
Sinh viên thực hiện : TRẦN TUẤN LINH

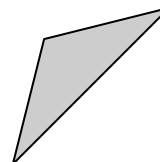
TP HỒ CHÍ MINH – NĂM 2018

## **LỜI CẢM ƠN**

Tôi xin cảm ơn nhà trường và các thầy cô khoa Công Nghệ Thông Tin đã hỗ trợ và tạo điều kiện để tôi thực hiện luận văn này. Cũng như gửi lời cảm ơn tới cô Ngô Thị Bảo Trân đã giúp đỡ, hướng dẫn để đạt được mục đích bài luận văn này. Đồng thời cũng gửi lời cảm ơn đến BlockStack Việt Nam đã hỗ trợ và giúp đỡ trong thời gian tìm hiểu trong việc phát triển ứng dụng. Cảm ơn công ty Ant-Tech đã cho nghỉ phép một thời gian dài để có thể thực hiện tốt luận văn. Cảm ơn anh Duy Hưng từ Umbala có những chia sẻ về blockchain cũng như ý tưởng để thực hiện sản phẩm này.

Ngày 9 tháng 7 năm 2018

Trần Tuấn Linh



# MỤC LỤC

Chương 1. GIỚI THIỆU .....	1
I. MỤC ĐÍCH LÀM LUẬN VĂN .....	1
Chương 2. KIẾN THỨC NỀN TẢNG .....	4
I. GIỚI THIỆU VỀ MÃ HOÁ.....	4
I.1 Mã hoá đối xứng.....	4
I.2 Mã hoá bất đối xứng.....	5
II. HÀM BẮM .....	7
II.1 Chữ kí điện tử (Digital Signature).....	8
Chương 3. GIỚI THIỆU BLOCKCHAIN VÀ HỆ THỐNG BITCOIN .....	9
I. ĐỊNH NGHĨA.....	9
II. HỆ THỐNG BITCOIN.....	9
II.1 Giới thiệu .....	9
III. MÔ TẢ KỸ THUẬT .....	11
III.1 Mạng Bitcoin (Bitcoin Network) .....	11
III.2 Ví (Wallet) .....	12
III.3 Giao dịch (Transaction).....	15
III.4 Đào và đồng thuận.....	18
III.5 An toàn bitcoin.....	25
Chương 4. GIỚI THIỆU ETHEREUM .....	27
I. ETHEREUM LÀ GÌ?.....	27
II. SỰ KHÁC NHAU GIỮA BITCOIN VÀ ETHEREUM .....	28
II.1 Smart Contract và Scripting.....	28
II.2 Accounts và UTXO .....	30
II.3 Thời gian tạo khối mới .....	31
Chương 5. ỨNG DỤNG GỌI VỐN TRÊN NỀN TẢNG ETHEREUM.....	32
I. GIỚI THIỆU .....	32
II. ỨNG DỤNG GỌI VỐN LIGHTHOUSE.....	33
III. VAI TRÒ .....	35
IV. CÔNG NGHỆ SỬ DỤNG.....	37

V. HÌNH ẢNH.....	39
Chương 6. TỔNG KẾT.....	43
I. ƯU NHƯỢC ĐIỂM .....	43
II. KẾT QUẢ ĐẠT ĐƯỢC .....	44
III. HƯỚNG PHÁT TRIỂN .....	45

# MỤC LỤC CÁC HÌNH VẼ

Hình 2-1: Mô hình mã hóa và giải mã mã hóa một chiều.....	5
Hình 2-2: Mô hình mã hóa và giải mã mã hóa bất đối xứng. ....	6
Hình 2-3: Mô hình hàm băm mật mã.....	8
Hình 3-1: Mô hình Base58Check .....	15
Hình 3-2: Danh sách tỉ lệ các blockchain có thể bị tấn công. ....	24
Hình 4-1: UTXO trong Bitcoin và Accounts trong Ethereum. ....	30
Hình 4-2: Externally owned và Contract accounts.....	31
Hình 5-1: Danh sách các blockchain có khả năng bị tấn công 51%.....	33
Hình 5-2: Sơ đồ chức năng của người dùng.....	36
Hình 5-3: Sơ đồ chức năng của thành viên .....	37
Hình 5-4: Sơ đồ chức năng của người sở hữu Lighthouse.....	37
Hình 5-5: Trang chủ của Lighthouse hiển thị các dự án.....	40
Hình 5-6: Trang hiển thị thông tin chi tiết của dự án .....	40
Hình 5-7: Trang quản lí thông tin cá nhân .....	41
Hình 5-8: Trang tạo dự án .....	41
Hình 5-9: Trang đăng kí thành viên.....	42

# Chương 1. GIỚI THIỆU

## I. MỤC ĐÍCH LÀM LUẬN VĂN

Tiền giấy là một phương tiện thanh toán phổ biến nhất hiện nay bên cạnh vàng và trái phiếu. Tuy nhiên tiền giấy cũng có những hạn chế như, dễ bị làm giả, bị hư hại theo thời gian. Những nhà phát hành tiền giấy luôn phải đấu tranh với vấn nạn tiền giả bằng việc tăng độ phức tạp của công nghệ giấy và in. Các loại tiền vật lí có thể dễ dàng giải quyết được vấn đề chi hai lần – khi mà một đơn vị tiền tệ được chi nhiều hơn một lần – vì tiền giấy không thể ở một lúc hai nơi. Khi đó, tiền điện tử ra đời giúp cho việc thanh toán trở nên nhanh chóng hơn. Nhưng tiền điện tử lại có những hạn chế như: độ bảo mật không tốt, chi phí giao dịch cao, thời gian giao dịch lâu và mọi hoạt động giao dịch đều phải thông qua một trung tâm xử lý trung gian

Vào cuối những năm 1980, khi mà mật mã đã bắt đầu trở nên phổ biến, các nhà nghiên cứu đã cố gắng sử dụng mật mã để xây dựng tiền điện tử. Những dự án tiền điện tử đầu tiên đã được phát hành sau đó. Tuy nhiên, những đồng tiền đầu tiên vẫn phải giao dịch thông qua một trung tâm thanh toán bù trừ tương tự như các ngân hàng truyền thống.

Để có thể chống lại các tác nhân có thể gây tổn hại, một hệ thống phải thực sự phi tập chung. Blockchain chính là một hệ thống như vậy, không một cơ quan trung ương hay cá nhân nào có thể kiểm soát hay tấn công được. Blockchain là một cấu trúc dữ liệu được xếp theo thứ tự, là một danh sách liên kết đuôi các block chứa các giao dịch. Blockchain có thể được lưu trong các tập tin hoặc trong một cơ sở dữ liệu đơn giản. Mỗi block được định danh bằng một hàm băm mật mã. Block sau sẽ chứa định danh của block trước. Mỗi block chỉ có thể nối tới duy nhất một block cha, nhưng block cha có thể có thể tạm thời có nhiều block con. Khi dữ liệu của block cha bị thay đổi dẫn đến định danh cũng thay đổi theo thì những block con nối phía sau cũng sẽ bị thay đổi, làm cho toàn bộ chuỗi đó trở nên không hợp lệ. Thêm nữa, khi thay đổi định danh thì cần phải tính toán lại định danh của chuỗi thì phải cần đến một sức mạnh tính toán vô cùng lớn. Do đó, một block càng nằm sâu trong Blockchain càng khó bị thay đổi.

Vào năm 2008, Satoshi Nakamoto đã phát minh ra Bitcoin - ứng dụng đầu tiên của Blockchain. Bằng cách kết hợp các công nghệ khác nhau để tạo nên một hệ thống tiền tệ hoàn toàn phi tập trung được gọi là bitcoin mà không cần dựa vào bất cứ tổ

chức nào để phát hành, thanh toán hoặc xác thực các giao dịch. Một phát minh quan trọng của Bitcoin đó là đã sử dụng hệ thống máy tính toán phân tán (thuật toán Proof-of-Work). Toàn hệ thống sẽ tiến hành một cuộc “bỏ phiếu” toàn cầu mỗi 10 phút, cho phép cả hệ thống đi đến thống nhất về trạng thái của các giao dịch sau đó các giao dịch đã được xác thực bởi mạng lưới sẽ được đưa vào một block và block này sẽ được thêm vào vào Blockchain. Với cách này đã giải quyết được vấn đề chi hai lần khi mà các loại tiền điện tử trước đây không thể giải quyết được. Bitcoin chính thức hoạt động từ năm 2009 cho đến nay. Là sự khởi đầu của kỷ nguyên Blockchain. Nhưng tiềm năng của Blockchain không chỉ dừng lại ở việc xây dựng hệ thống tiền điện tử. Tiềm năng của nó còn vượt xa hơn thế. Vào cuối năm 2013, một nền tảng Blockchain mới ra đã ra đời. Đó là Ethereum, một cơ sở hạ tầng điện toán phi tập trung toàn cầu cho phép thực thi các chương trình được gọi là Smart Contract. Ethereum sử dụng Blockchain để đồng bộ và lưu trữ trạng thái của hệ thống cùng với một đơn vị tiền tệ được gọi là ether là một thước đo và hạn chế các tài nguyên được thực thi.

Smart Contract là những chương trình máy tính. Không có nghĩa pháp lý trong ngữ cảnh này. Một khi được triển khai thì không thể thay đổi nội dung đoạn mã bên trong được nữa, cách duy nhất là phải triển khai một Smart Contract mới. Kết quả thực thi của mọi smart contract đều giống nhau cho bất cứ ai thực thi nó trong ngữ cảnh của giao dịch và trạng thái của Blockchain trong thời điểm thực thi. Bởi vì trạng thái khởi tạo và kết quả đều giống giống nhau tại mọi nút trong mạng lưới nên toàn bộ hệ thống hoạt động như một máy tính duy nhất trên thế giới.

KickStarter cho phép những nhà kinh doanh có khả năng phát triển sản phẩm trình bày dự án của mình, nhằm gọi vốn từ người dùng trên Kickstarter ở phạm vi toàn cầu. Khi một dự án được đưa lên Kickstarter để kêu gọi vốn, dự án bắt buộc phải xác định mức vốn đầu tư cần có và thời gian thực hiện chiến dịch gọi vốn cho dự án. Khi dự án thành công, Kickstarter thu 5% trên tổng số tiền huy động được, số tiền còn lại sẽ được chuyển cho chủ dự án. Nếu dự án không thành công, tiền ủng hộ sẽ được hoàn trả cho chủ đầu tư. Chủ dự án bắt buộc phải thực hiện dự án sau khi nhận tiền và trả lãi hoặc phần thưởng theo thỏa thuận ban đầu cho nhà đầu tư. Vấn đề ở đây là chúng ta hoàn toàn phụ thuộc vào KickStarter, thông tin về khách hàng, dự án đều được thu thập, mức tính phí khá cao có thể sẽ ảnh hưởng đến dự án. Với 5% trên tổng số tiền, thêm 3 – 5% các chi phí chuyển tiền, pháp lý khác. Với Blockchain - ở đây là smart contract có thể giải quyết được vấn đề này. Với khả năng chuyển tiền trực tiếp một cách nhanh chóng, ẩn danh và chi phí thấp sẽ không làm ảnh hưởng đến tiến độ dự án mà vẫn đảm bảo tính riêng tư, bảo mật. Một khi smart contract



được triển khai thì sẽ không thể thay đổi được, đảm bảo sự minh bạch trong quá trình gây quỹ.

Cuốn báo cáo này được chia thành bảy chương với nội dung như sau:

- Chương 1: Giới thiệu.
- Chương 2: Kiến thức nền tảng.
- Chương 3: Giới thiệu về Blockchain.
- Chương 4: Giới thiệu về nền tảng Ethereum.
- Chương 5: Ứng dụng gọi vốn trên nền tảng Ethereum.
- Chương 6: Tổng kết.

# Chương 2. KIẾN THỨC NỀN TẢNG

## I. GIỚI THIỆU VỀ MÃ HOÁ

Internet là một môi trường mở, những thông tin bạn gửi lên internet hoặc nhận về từ internet đều có thể bị nghe trộm, thay đổi. Do đó việc bảo mật những thông tin này là hết sức cần thiết, một trong những cách để bảo mật thông tin hữu hiệu nhất hiện nay là mã hoá. Mã hoá là một thứ cực kì quan trọng, và có mặt tại rất nhiều nơi trong đời sống hàng ngày của chúng ta. Nếu không có mã hoá, hệ thống ATM sẽ không tồn tại, sẽ không tồn tại chuỗi hệ thống ngân hàng, không có thương mại điện tử, internet sẽ không phát triển. Mã hoá là một phương pháp để bảo vệ thông tin, bằng cách chuyển đổi thông tin từ dạng rõ (thông tin ở trạng thái bình thường – *plain text*) sang dạng mờ (thông tin đã bị che đi, nên không thể hiểu được – *cipher text*). Có nhiều mô hình mã hoá khác nhau. Mỗi loại có những ưu và nhược điểm riêng. Tuy nhiên bất kỳ mô hình nào cũng đều trải qua ba giai đoạn cơ bản:

- **Tạo khoá** (*Key generator*): tạo ra bộ khóa sử dụng trong quá trình mã hóa và giải mã.
- **Mã hoá** (*Encryption*): biến đổi thông điệp gốc thành thông điệp mã hóa.
- **Giải mã** (*Decryption*): giải mã thông điệp mã hóa thành thông điệp gốc ban đầu.

Ta có thể chia ra các phương pháp mã hoá thành hai loại chính:

- Mã hoá đối xứng. (*symmetric encryption*)
- Mã hoá bất đối xứng. (*asymmetric encryption*)

Trong phạm vi đề tài này chỉ đề cập đến mã hoá đối xứng và bất đối xứng.

### I.1 Mã hoá đối xứng

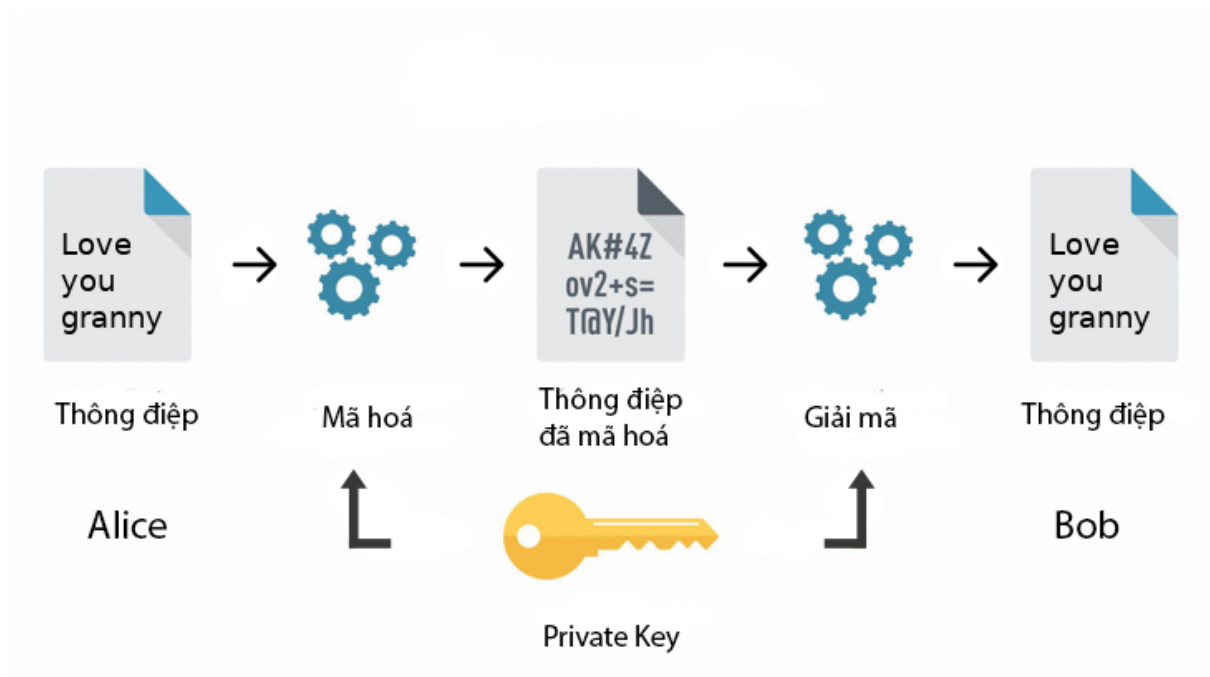
Trong mô hình mã hoá đối xứng, một khoá bí mật (*private key*) được dùng trong cả quá trình mã hoá và giải. Gọi *Gen*, *Enc*, *Dec* lần lượt là thuật toán tạo khoá, mã hoá và giải mã. Thông điệp là *m*, thông điệp đã mã hóa là *c* và khoá bí mật *k*.

Mô hình mã hóa đối xứng

- **Tạo khoá:** nhận giá trị input là  $1^n$  (tham số an toàn) và output là khoá bí mật  $k$ .
- **Mã hoá:** nhận giá trị input là một khoá  $k$  cùng với thông điệp  $m$ , output là ciphertext  $c$ .
- **Giải mã:** nhận giá trị output là một khoá  $k$  cùng với ciphertext  $c$ , input là thông điệp  $m$ .

*Bài toán:* Alice muốn gửi một thông điệp cho Bob qua Internet mà không muốn bị lộ thông tin cho người khác có thể nhìn thấy.

*Lời giải:* Alice và Bob quyết định sử dụng mô hình mã hoá đối xứng, trong đó khóa bí mật được thống nhất từ trước để mã hoá thông điệp gửi đi. Ta có mô hình sau:



Hình 2-1: Mô hình mã hóa và giải mã mã hóa một chiều.

Một số mô hình mã hóa đối xứng thường gặp là: Triple DES, DES, AES,...

## I.2 Mã hoá bất đối xứng

Mô hình mã hoá bất đối xứng sử dụng một cặp khoá bao gồm: khoá công khai (*public key*) và khoá bí mật (*private key*). Khoá công khai được dùng để mã hoá và khoá bí mật được dùng để giải mã.

Với *Gen*, *Enc*, *Dec* lần lượt là thuật toán tạo khoá, mã hoá và giải mã. Thông điệp là  $m$ ,  $c$  là thông điệp đã được mã hoá. Khoá bí mật kí mật là  $sk$  và khoá công khai là  $pk$ .

Tương tự như các bước như mã hoá đối xứng, nhưng có vài sự khác biệt:

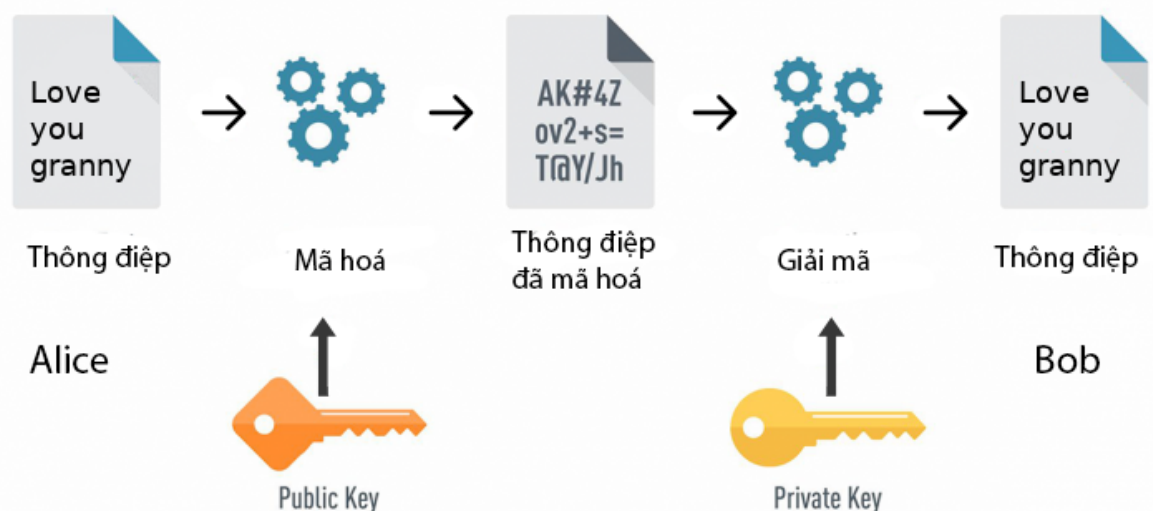
Mô hình mã hóa đối xứng

- **Tạo khoá:** nhận giá trị input là  $1^n$ , output là là một cặp khoá  $(pk, sk)$ .
- **Mã hoá:** nhận giá trị input là một khoá  $pk$  cùng với thông điệp  $m$ , output là ciphertext  $c$ .
- **Giải mã:** nhận giá trị input là một khoá  $sk$ , cùng với ciphertext  $c$ , input là thông điệp  $m$ .

*Bài toán.* Tương tự như mô hình mã hóa đối xứng, Alice muốn gửi một thông điệp cho Bob qua Internet mà không muốn bị lộ thông tin cho người khác có thể nhìn thấy.

*Lời giải.* Alice và Bob quyết định sử dụng mô hình mã hoá đối xứng. Bob sở hữu một bộ khóa  $(pk, sk)$ . Khóa công khai  $pk$  được công bố rộng rãi cho tất cả mọi người bao gồm cả Alice. Khóa bí mật  $sk$  được Bob giữ bí mật. Để gửi thông điệp  $m$  cho Bob, Alice sử dụng khóa công khai  $pk$  để mã hóa  $m$  thành ciphertext  $c$  và gửi qua cho Bob. Bob sử dụng khóa bí mật  $sk$  để giải mã  $c$  thành  $m$  như ban đầu.

Ta có mô hình sau:



Hình 2-2: Mô hình mã hóa và giải mã mã hóa bất đối xứng.

Điểm yếu lớn nhất của mã hóa bất đối xứng là tốc độ mã hóa và giải mã rất chậm so với mã hóa đối xứng, nếu dùng mã hóa bất đối xứng để mã hóa dữ liệu truyền – nhận giữa hai bên thì sẽ tốn rất nhiều chi phí. Do đó, mô hình mã hoá bất đối xứng thường được dùng để trao đổi khoá bí mật, khoá này sẽ được dùng để mã hoá dữ liệu trong mô hình mã hoá đối xứng.

Một số mô hình mã hoá bất đối xứng thường thấy là: RSA và Elliptic Curve Cryptography (ECC). Trong đó mô hình ECC là mô hình được sử dụng phổ biến trong các hệ thống blockchain hiện nay.

## II. HÀM BẮM

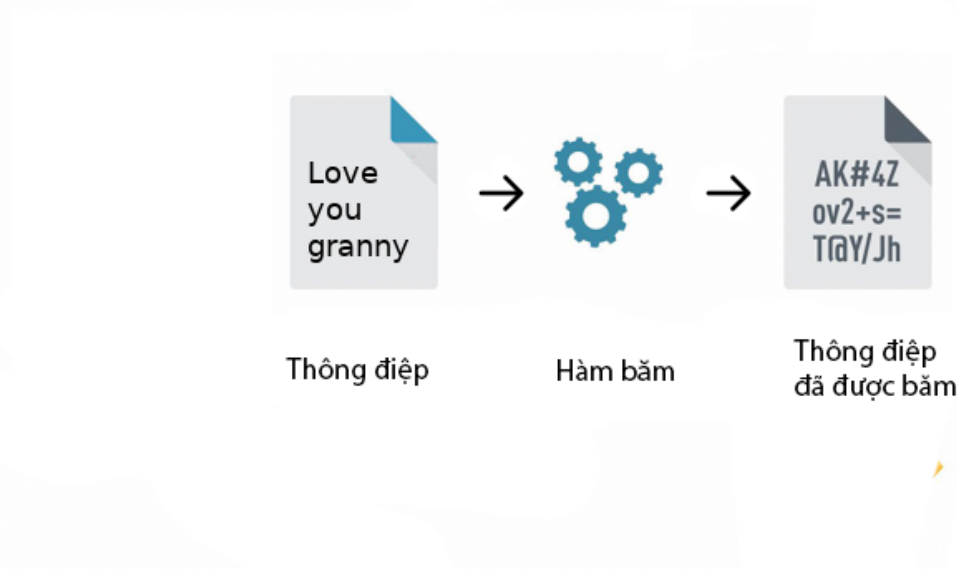
Hàm băm (*hash function*) được dùng để chuyển đổi một thông điệp hoặc dữ liệu với kích thước tùy ý thành một thông điệp có kích thước cố định. Hàm băm là hàm một chiều, không thể đảo ngược lại thông điệp gốc từ thông điệp băm được.

Một hàm băm bất kì phải có các đặc tính sau:

- Cùng một thông điệp luôn tạo ra cùng một hash.
- Không thể đảo ngược lại giá trị ban đầu.
- Không thể có hai giá trị khác nhau có cùng một hash.
- Tốc độ của hàm băm phải nhanh.

Một vài công dụng của hàm băm:

- Kiểm tra tính toàn vẹn của dữ liệu.
- Tạo các giá trị ngẫu nhiên.
- Được dùng trong các lược đồ chữ kí điện tử.



Hình 2-3: Mô hình hàm băm mật mã.

## II.1 Chữ kí điện tử (Digital Signature)

Chữ kí điện tử là một kỹ thuật xác thực cho phép người tạo ra nội dung của một thông điệp được quyền đính kèm một đoạn dữ liệu số như là chữ kí đánh dấu của người chủ với nội dung đã tạo ra. Chữ kí số được tạo ra bằng cách băm (hash) nội dung thông điệp thành một thông điệp mới với kích thước nhỏ hơn rất nhiều so với thông điệp gốc, sau đó người dùng sẽ (ký) mã hoá thông điệp vừa băm với khoá bí mật của mình. Để xác thực thông điệp vừa được ký có phải là của người chủ thông điệp hay không, người ta tiến hành giải mã thông điệp đã được băm, băm thông điệp gốc một lần nữa và so sánh xem hai thông điệp vừa nhận được có giống nhau hay không.

Chữ kí số phải đảm bảo những tính chất sau đây:

- Tính xác thực (*Authenticity*): dùng để chứng thực nguồn gửi nội dung thông điệp. Giúp người nhận chứng thực được ai đã gửi thông điệp.
- Tính toàn vẹn (*Integrity*): dùng để kiểm tra tính toàn vẹn dữ liệu của nội dung được gửi đi là không bị thay đổi hay chỉnh sửa kể từ lúc tạo chữ kí số vào văn bản được gửi đi.
- Chống thoái thác (*Non-repudiation*): giúp người nhận khi kiểm tra nội dung đã được kí bởi chữ kí điện tử kèm theo. Đảm bảo chắc chắn người kí không thể chối cãi về những gì mình đã tạo ra ở thời điểm bắt đầu tạo chữ kí số.

# Chương 3. GIỚI THIỆU BLOCKCHAIN VÀ HỆ THỐNG BITCOIN

## I. ĐỊNH NGHĨA

Blockchain là một cấu trúc dữ liệu được xếp theo thứ tự, là một danh sách liên kết đuôi các block chứa các giao dịch. Blockchain có thể được lưu trong như một tập tin hoặc trong một cơ sở dữ liệu đơn giản. Các block được liên kết ngược trở lại với block trước đó trong chuỗi. Mỗi block trong blockchain được xác định bằng một địa chỉ và cách tạo ra địa chỉ bằng hàm băm mật mã SHA256. Mỗi block tham chiếu tới block trước đó, gọi là block cha, thông qua địa chỉ.

Mỗi block con chỉ có một block cha. Một block cha có thể có tạm thời nhiều block con. Mỗi block con này tham chiếu tới cùng một block cha và chứa cùng một mã băm của block cha. Mã băm của block cha thay đổi sẽ làm liên kết của block con đến block cha cũng thay đổi theo. Điều này làm cho các block theo sau nó cũng phải thay đổi theo và quá trình tính toán lại như vậy đòi hỏi một block lượng tính toán khổng lồ nên rất tốn kém để thực hiện, nên chuỗi dài nhất khiến cho blockchain có tính bất biến.

Một blockchain mở, công khai phải có các thành phần cơ bản sau đây:

- Một cơ sở dữ liệu phi tập trung, lưu lại toàn bộ quá trình thay đổi trạng thái của hệ thống.
- Một mạng lưới ngang hàng kết nối những người tham gia và lan truyền các giao dịch và các block chứa các giao dịch đã được xác nhận.
- Một bộ trạng thái, dưới hình thức của các giao dịch.
- Một bộ quy tắc đồng thuận (*Consensus rules*)
- Một thuật toán đồng thuận (ví dụ: *Proof-of-Work*, *Proof-of-Stack*, ...)

## II. HỆ THỐNG BITCOIN

### II.1 Giới thiệu

Hệ thống Bitcoin ra đời năm 2008 cùng với một tài liệu có tựa đề là “Bitcoin: A Peer-to-Peer Electronic Cash System” được công bố bởi Satoshi Nakamoto. Mạng Bitcoin được khởi động từ năm 2009. Hệ thống Bitcoin gồm hai loại tiền tệ là bitcoin

và satoshi, một bitcoin bằng 100.000.000 satoshi. Người dùng có thể tích trữ và trao đổi giá trị với nhau. Từ đây hệ thống Bitcoin sẽ được kí hiệu là Bitcoin và bitcoin để chỉ đồng tiền sử dụng trên hệ thống này.

Người dùng có thể giao dịch bitcoin qua mạng lưới và có thể thực hiện hầu như mọi việc tương tự với tiền tệ truyền thống, bao gồm mua bán hàng hóa, chuyển tiền đến các cá nhân hay tổ chức hoặc cho vay tín dụng. Ngoài ra, người dùng có thể mua bán trao đổi bitcoin với các loại tiền tệ khác trên những sàn giao dịch chuyên dụng như: Coinbase, Benance, .... Có thể xem bitcoin là mẫu hình tiền tệ hoàn hảo cho Internet trong tương lai. Không như tiền tệ truyền thống, bitcoin hoàn toàn ảo. Không có đồng tiền bitcoin vật lý nào. Bitcoin được ngầm định trong các giao dịch chuyển giao giá trị từ người gửi đến người nhận. Người dùng bitcoin sở hữu bộ khóa cho phép họ chứng minh quyền sở hữu bitcoin trên mạng lưới Bitcoin và có thể sử dụng bộ khóa này để xác nhận giao dịch và chuyển tiền cho người khác. Các khóa thường được lưu trữ trong một ví trên máy tính hoặc điện thoại của người dùng. Cách duy nhất để chi tiêu bitcoin là người dùng phải sở hữu bộ khóa để có thể xác minh lên các giao dịch gửi và nhận, điều này trao lại quyền kiểm soát hoàn toàn vào tay người dùng.

Hệ thống Bitcoin là một hệ thống phân tán ngang hàng, do đó không có bất kì một máy chủ trung tâm nào. Tiền bitcoin được tạo ra thông qua một quá trình được gọi là “đào”, trong đó các thợ đào phải cạnh tranh với nhau để tìm đáp án cho một bài toán được đặt ra bởi mạng lưới bitcoin. Bất kỳ người tham gia nào trên mạng lưới bitcoin cũng đều có thể trở thành thợ đào, sử dụng sức mạnh tính toán của máy tính của mình để xử lý, xác minh và ghi nhận các giao dịch. Trung bình, cứ mỗi 10 phút, một thợ đào bitcoin có thể xác thực những giao dịch của 10 phút trước đó và được thưởng một lượng bitcoin vừa được tạo ra. Việc đào bitcoin đã phi tập trung hóa các chức năng phát hành tiền tệ và thanh toán bù trừ của một ngân hàng trung ương.

Giao thức bitcoin tích hợp các thuật toán để điều tiết chức năng đào bitcoin trên mạng lưới. Độ khó của tác vụ xử lý mà các thợ đào phải thực hiện được điều chỉnh liên tục sao cho trung bình cứ sau 10 phút lại có một thợ đào tìm được đáp án cho bài toán mà mạng lưới đưa ra, bất kể có bao nhiêu thợ đào đang tham gia cạnh tranh tìm đáp án tại bất kỳ thời điểm nào. Sau mỗi 4 năm, giao thức bitcoin cũng giảm một nửa tốc độ tạo bitcoin mới, số lượng bitcoin tối đa có thể được tạo ra cố định ở mức 21 triệu bitcoin. Ước tính toàn bộ số bitcoin sẽ được khai thác hết vào năm 2140. Do tốc độ phát hành bitcoin giảm dần theo thời gian nên không thể khiến bitcoin lạm phát bằng cách tạo thêm bitcoin.



Một cấu trúc dữ liệu blockchain cơ bản gồm hai phần header và body. Phần header lưu thông tin của block và phần body lưu các giao dịch.

Cấu trúc cơ bản của một block như sau:

```
1. {
2.   "size": 43560,
3.   "version": 2,
4.   "previousblockhash": "0000000000000027e7ba6fe7bad39faf3b5a83daed765f05f7d1b71a163
   2249",
5.   "merkleroot": "5e049f4030e0ab2debb92378f53c0a6e09548aea083f3ab25e1d94ea1155e29d",
6.   "time": 1388185038,
7.   "difficulty": 1180923195.2580261,
8.   "nonce": 4215469401,
9.   "tx": [
10.    "257e7497fb8bc68421eb2c7b699dbab234831600e7352f0d9e6522c7cf3f6c77",
11.    "05cfd38f6ae6aa83674cc99e4d75a1458c165b7ab84725eda41d018a09176634"
12.  ]
13. }
```

Trong đó có ba tham số quan trọng là:

- previousblockhash là id của khối trước đó.
- difficulty là độ khó tại thời điểm khối đó được tạo ra.
- tx là một mảng chứa id các giao dịch.

Đằng sau đó, Bitcoin cũng là tên của một giao thức, một mạng ngang hàng và là một mạng lưới điện toán phân tán. Đồng tiền bitcoin cơ bản chỉ là một ứng dụng đầu tiên của phát minh này. Bitcoin là sự kết hợp độc đáo và mạnh mẽ kết hợp từ:

- Một mạng ngang hàng phi tập trung (*Peer-to-peer*)
- Một sổ cái giao dịch công khai (*Blockchain*)
- Một tập hợp các quy tắc để xác thực các giao dịch và phát hành tiền tệ độc lập (các quy tắc đồng thuận, *Consensus rules*)
- Một cơ chế để đạt được sự đồng thuận phi tập trung trên blockchain hợp lệ (thuật toán bằng chứng công việc, *Consensus algorithm*)

### III. MÔ TẢ KỸ THUẬT

#### III.1 Mạng Bitcoin (Bitcoin Network)

Bitcoin được xây dựng như một kiến trúc mạng ngang hàng (*P2P*) dựa trên Internet. Trong mạng này không có máy chủ, không có các dịch vụ tập trung. Các nút trong mạng P2P vừa là máy chủ vừa là máy khách.

Mặc dù các nút trong mạng P2P Bitcoin là bình đẳng nhưng chúng có thể đóng nhiều vai trò khác nhau tùy theo tính năng mà chúng hỗ trợ. Một nút bitcoin là tập hợp các chức năng: định tuyến, cơ sở dữ liệu blockchain, đào và các dịch vụ ví.

Tất cả các nút đều có chức năng định tuyến để tham gia vào mạng lưới và có thêm các tính năng khác. Tất cả các nút đều xác thực, phát tán các giao dịch và block, phát hiện và duy trì kết nối tới các nút khác. Một số nút chỉ duy trì một tập con của blockchain và xác minh các giao dịch bằng một phương pháp được gọi là “xác minh thanh toán giản lược” hay SPV (*Simplified Payment Verification*).

Một số loại nút khác nhau trên mạng bitcoin mở rộng:

- Phần mềm tham chiếu (*Bitcoin Core*): Chứa một ví, thợ đào, cơ sở dữ liệu blockchain đầy đủ và nút định tuyến mạng trên mạng P2P Bitcoin.
- Nút blockchain đầy đủ (*Full Blockchain Node*): Chứa một cơ sở dữ liệu blockchain đầy đủ và nút định tuyến trên mạng P2P Bitcoin. Các nút đầy đủ duy trì một bản sao hoàn chỉnh và cập nhật thường xuyên của blockchain. Các nút đầy đủ có khả năng độc lập xác nhận bất kì giao dịch nào mà không cần nhờ vào một bên thứ ba tin cậy.
- Thợ đào riêng lẻ (*Solo Miner*): Chứa một chức năng đào với bản sao blockchain đầy đủ và một nút định tuyến mạng P2P bitcoin.
- Ví rút gọn (*Lightweight wallet*): Chứa một ví và một nút mạng trên giao thức P2P bitcoin, không chứa blockchain.

Khi một nút mới khởi động, để có thể tham gia vào mạng bitcoin, nó cần phải phát hiện các nút bitcoin khác trên mạng lưới. Để bắt đầu quá trình này nó cần phải tìm được và kết nối với ít nhất một nút đang tồn tại trên mạng lưới. Các nút được chọn ngẫu nhiên bất kể vị trí địa lý.

Khi một nút mới hoàn toàn khởi động thì nó sẽ kết nối tới một số nút mặc định đã được thiết lập sẵn trong bản thực thi bitcoin. Các nút trong mạng lưới sẽ tiếp tục khám phá các nút mới và ngừng kết nối với các nút cũ, cũng như hỗ trợ các nút khác khởi động. Trong vòng 90 phút nếu một nút không kết nối được tới bất kì một nút nào mà nó đang kết nối thì sẽ xem như nút đó đã tắt kết nối và tìm tới nút mới.

### III.2 Ví (Wallet)

Ví trong bitcoin không hề chứa bất kì bitcoin nào. Về cơ bản, ví là một phần mềm dùng để theo dõi số dư, quản lý các khóa, địa chỉ, tạo và ký các giao dịch. Ví chứa các khóa, thông qua các khóa đó người dùng kiểm soát được toàn bộ số tiền của mình được lưu trên blockchain tương ứng với các khóa đó. Người dùng ký các giao dịch

bằng các khóa đó, qua đó chứng minh rằng họ là chủ sở hữu của các giao dịch. Ví có mặt hầu hết trên các nền tảng phổ biến khác nhau như Mac OS, Windows, Linux, Android, iOS, ... Có hai loại ví chính bao gồm: ví bất định (*Nondeterministic*) và ví tất định (*Deterministic*)

### 2.1 Ví bất định

Trước đây, các ví thường tạo ra một số lượng ngẫu nhiên các khóa bí mật. Ví dụ, trong bản thực thi Bitcoin Core ban đầu tạo sẵn 100 khóa bí mật ngẫu nhiên và mỗi khóa này chỉ được sử dụng một lần. Khi hết thì sẽ tiếp tục tạo mới, nên việc quản lý các khóa này vô cùng khó khăn, có thể dẫn đến mất số bitcoin vĩnh viễn nếu làm mất khóa.

### 2.2 Ví tất định

Do những hạn chế của ví bất định, nên một loại ví tất định ra đời từ đó. Ví tất định là tập hợp các khóa được tạo ra từ một hạt giống (seed) chung. Hạt giống này được tạo một cách ngẫu nhiên sau đó kết hợp với các dữ liệu khác. Chỉ cần hạt giống này là có thể tạo lại tất cả các khóa trong ví. Do đó chỉ cần một bản sao lưu vào đúng thời điểm khởi tạo là đủ.

### 2.3 Ví HD (Hierarchical deterministic)

Các ví tất định mới dựa trên các tiêu chuẩn mới còn cho phép chứa các khóa theo cấu trúc hình cây, một khóa cha có thể có nhiều khóa con và từ mỗi khóa con có thể có nhiều khóa cháu, cứ thế diễn ra cho tới vô tận.

Ưu điểm thứ nhất của loại ví HD này so với ví tất định thông thường là cho phép dễ dàng biểu đạt ý nghĩa về mặt tổ chức, có thể dùng một nhánh con để nhận các giao dịch trong khi một nhánh khác dùng để nhận tiền thừa trả lại.

Ưu điểm thứ hai là người dùng có thể dùng một khóa công khai mà không cần truy cập vào khóa bí mật tương ứng. Cho phép tạo hoặc đặt tạo các khóa ở những môi trường không an toàn như trên các máy chủ.

### 2.4 Khóa (Key)

Khóa, địa chỉ và chữ ký số được sử dụng để thiết lập quyền sở hữu bitcoin. Các khóa này không được lưu trên mạng lưới mà do người dùng tạo ra và lưu trong một tập tin hay một cơ sở dữ liệu đơn giản được gọi là ví. Các khóa này được tạo hoàn toàn độc lập mà không cần phải kết nối tới blockchain.

Để các giao dịch được thêm vào blockchain hầu hết các giao dịch phải có chữ ký số hợp lệ và phải được tạo ra bằng khóa bí mật. Do đó, nếu ai đó có được khóa bí mật sẽ có thể kiểm soát được toàn bộ bitcoin.

Các khóa đi theo cặp, bao gồm khóa bí mật và khóa công khai. Khóa công khai được dùng để nhận các giao dịch còn khóa bí mật dùng để tạo ra chữ ký số để có thể chi tiêu các giao dịch này.

### 2.5 Địa chỉ (Address)

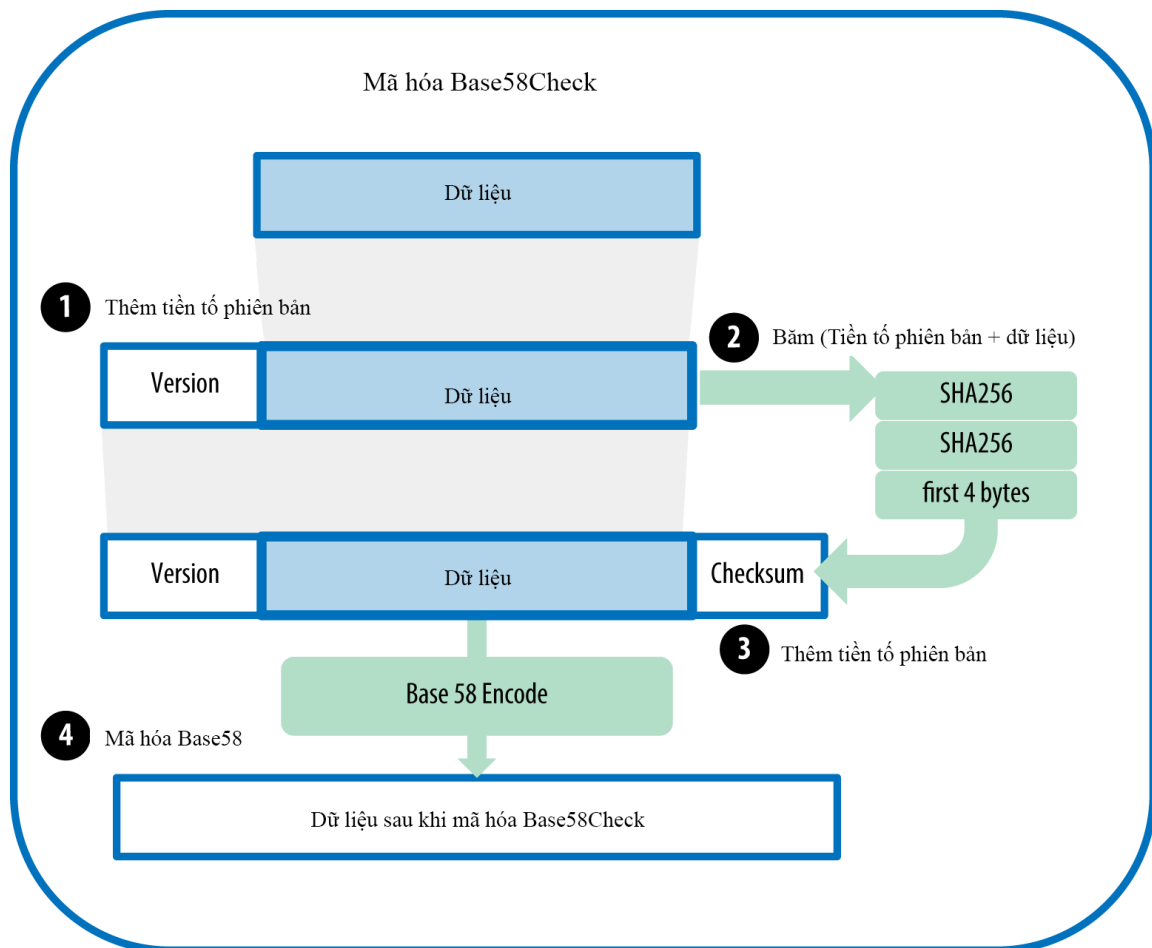
Địa chỉ bitcoin là một dãy các chữ số và ký tự được dùng để chia sẻ với bất kì ai muốn chuyển tiền cho bạn. Các địa chỉ thường được tạo ra từ các khóa công khai thường bắt đầu với số “1”. Dưới đây là một địa chỉ bitcoin được tạo ra từ khóa công khai:

1LbURpSh1jMxtzMensUHbztAELJXXJ9vub

### 2.6 Mã hóa Base58 và Base58Check

Để biểu diễn một cách ngắn gọn các con số lớn, các hệ thống máy tính thường dùng các hệ số lớn hơn cơ số 10. Hệ cơ số 58 sử dụng các chữ cái viết hoa, viết thường và các số từ 0 đến 9 để biểu diễn, nhưng sẽ bỏ đi một số ký tự thường dễ bị nhầm lẫn do có cách hiển thị giống nhau trong một số font chữ. Trong cơ số 58 không có các ký tự 0 (số không), O (chữ o hoa), l (chữ L thường), I (chữ I hoa).

Để ngăn chặn việc gõ nhầm hay đọc nhầm, bitcoin đã sử dụng Base58Check. Base58Check là một dạng mã hóa Base58 được tích hợp thêm mã kiểm lỗi (checksum). Giá trị checksum này được tạo ra từ mã băm của dữ liệu mã hóa bao gồm tiền tố phiên bản và dữ liệu, sau đó lấy 4 byte đầu tiên thêm vào cuối dữ liệu đang được mã hóa. Tiền tố được dùng để dễ dàng xác định loại dữ liệu được mã hóa. Trong trường hợp dùng khóa công khai để tạo thành địa chỉ thì có tiền tố là 0.



Hình 3-1: Mô hình Base58Check

### III.3 Giao dịch (Transaction)

Có thể xem các giao dịch trong Bitcoin là phần quan trọng nhất. Các yếu tố khác trong Bitcoin đều được thiết kế nhằm đảm bảo rằng các giao dịch có thể được tạo ra, phát tán, xác thực và được thêm vào blockchain. Các giao dịch là sự chuyển giao giá trị giữa những người trong mạng lưới Bitcoin. Một giao dịch luôn bao gồm hai yếu tố cơ bản là *input* và *output*. Trong bitcoin thì output là cái có trước, tức là có những giao dịch chỉ có output nhưng không có input.

#### 3.1 Output

Mọi giao dịch đều tạo output, output là các block chứa tiền bitcoin không thể chia tách, được ghi vào blockchain và được mạng lưới công nhận là hợp lệ. Các nút Bitcoin đầu đủ theo dõi tất cả các output hiện có và có thể chi tiêu thường được gọi là “các đầu ra giao dịch chưa chi tiêu” hay UTXO (unspent transaction outputs).

Output của các giao dịch bao gồm hai phần:

- Một lượng bitcoin được tính mệnh giá bằng satoshi.

- Một câu đố mật mã quyết định các điều kiện cần có để có thể chi tiêu được output.

Câu đố mật mã này còn được gọi là kịch bản khóa, kịch bản nhân chứng hay scriptPubKey.

Khái niệm sổ tài khoản trên các ví là sự tổng hợp của tất cả các UTXO mà ví của người dùng đó có thể chi tiêu và các UTXO này sẽ nằm rải rác trong hàng trăm giao dịch của hàng trăm block khác nhau. Trong Bitcoin không hề có khái niệm sổ dư tài khoản.

Do bản chất không thể chia tách các giá trị output giao dịch. Hầu hết, khi chi tiêu một UTXO mà giá trị lớn hơn giá trị giao dịch mong muốn thì ta vẫn phải chi tiêu toàn bộ UTXO này và tiền thừa sẽ được trả lại trong giao dịch. Nếu giá trị giao dịch lớn hơn giá trị của một UTXO thì ví sẽ thu gom các UTXO khác cộng lại để có thể đạt được tổng giá trị phải lớn hơn hoặc bằng giá trị giao dịch mong muốn.

Một giao dịch có output cơ bản có cấu trúc như sau:

```
1. {
2.   "vout": [
3.     {
4.       "value": 0.015,
5.       "scriptPubKey": "OP_DUP OP_HASH160 ab68025513c3dbd2f7b92a94e0581f5d50f654e7 OP
   _EQUALVERIFY OP_CHECKSIG"
6.     },
7.     {
8.       "value": 0.0845,
9.       "scriptPubKey": "OP_DUP OP_HASH160 7f9b1a7fb68d60c536c2fd8aeaa53a8f3cc025a8 OP
   _EQUALVERIFY OP_CHECKSIG"
10.    }
11.  ]
12. }
```

Trong đó output là một mảng, trong đó chứa các giá trị cơ bản sau:

- value là giá trị bitcoin.
- scriptPubKey là câu đố để khóa giao dịch lại, chỉ khi cung cấp một scriptSig phù hợp thì mới có thể mở khóa được.

### 3.2 Input

Khi tạo một giao dịch, ví sẽ chọn ra trong số các UTXO mà nó đang kiểm soát điểm kiểm tra xem UTXO đó có đủ giá trị để thực hiện thanh toán được yêu cầu. Mỗi input được trỏ đến một UTXO tương ứng và phải chứa câu trả lời cho câu đố nằm trong output.

Một giao dịch có output cơ bản như sau:

```
1. {
2.   "vin": [
3.     {
4.       "txid": "7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18",
5.       "vout": 0,
6.       "scriptSig": "3045022100884d142d86652a3f47ba4746ec719bbfbd040a570b1deccbb6498c
75c4ae24cb02204b9f039ff08df09cbe9f6addac960298cad530a863ea8f53982c09db8f6e3813[ALL]0
484ecc0d46f1918b30928fa0e4ed99f16a0fb4fde0735e7ade8416ab9fe423cc5412336376789d172787
ec3457eee41c04f4938de5cc17b4a10fa336a8d752adf",
7.       "sequence": 4294967295
8.     }
9.   ]
10. }
```

Trong đó input là một mảng, trong đó chứa các giá trị cơ bản sau:

- txid là id của giao dịch nằm trong UTXO.
- vout là vị trí mà của output nằm trong giao dịch đó.
- scriptSig là chữ ký được cung cấp để mở khóa output tương ứng.
- sequence dùng để khóa giao dịch sau một khoảng thời gian hoặc một số block nhất định và giao dịch này sẽ được đào sau khi khoảng thời gian hoặc khối nhất định.

### 3.3 Phí giao dịch

Để các giao dịch có thể được thực hiện và thêm vào blockchain thì cần phải có các thợ đào để thực hiện các công việc này. Bất cứ ai cũng có thể sử dụng máy tính của mình để tiến hành đào các giao dịch và phần thưởng cho việc đào thành công là tiền bitcoin mới được tạo ra và phí giao dịch.

Hầu hết các giao dịch đều được tính phí để trả công cho các thợ đào bitcoin vì đã củng cố và xác thực các block cho mạng lưới Bitcoin. Phí đóng vai trò khuyến khích các thợ đào nhanh chóng đưa các giao dịch vào block tiếp theo, nếu phí thấp sẽ được xử lý chậm hơn hoặc thậm chí là bị loại ra khỏi mạng lưới. Phí cũng đóng vai trò là một cơ chế an ninh khi làm cho các cuộc tấn công trở nên tốn kém về mặt kinh tế.

Phí giao dịch được tính dựa trên kích thước giao dịch theo kilobyte chứ không phải dựa trên giá trị của giao dịch. Do trong input của một giao dịch chỉ trỏ đến UTXO nên sẽ không biết được bất cứ thông tin gì trong UTXO khi chưa tìm ra UTXO tương ứng trong danh sách các UTXO. Khi một giao dịch có nhiều input sẽ trỏ tới nhiều UTXO và các thợ đào phải tốn nhiều công sức hơn để tìm tất cả các UTXO đó để xác thực cho giao dịch đó nên phí sẽ cao hơn nhiều bất kể giá trị giao dịch đó giá trị bao nhiêu.

Trong một giao dịch không có trường nào để lưu mức phí của một giao dịch mà nó được ngầm định với công thức:

Phí = Tổng các output – Tổng các input

### III.4 Đào và đồng thuận

#### 4.1 Đào (Mining)

Nhờ việc đào mà sự bảo mật của Bitcoin được phi tập trung hóa và cho phép sự xuất hiện của sự đồng thuận toàn mạng lưới mà không cần phải có một cơ quan quyền lực trung ương nào. Mục đích của việc đào không phải là tạo ra bitcoin mới. Đó là hệ thống khuyến khích. Phần thưởng cho việc đào gồm các đồng bitcoin mới và phí giao dịch, đó là một mô hình khích lệ nhằm hỗ trợ hoạt động của các thợ đào vì sự bảo mật của mạng lưới, đồng thời phát hành tiền tệ.

Các thợ đào xác thực các giao dịch mới và ghi chúng vào blockchain. Trung bình mỗi block được đào trong vòng 10 phút. Các giao dịch sau khi trở thành một phần của block sẽ được thêm vào blockchain và được coi là “đã được xác nhận”, cho phép những chủ sở hữu mới bitcoin có thể tiêu bitcoin mà họ nhận được.

Để kiếm được phần thưởng này, các thợ đào phải cạnh tranh để giải quyết một vấn đề toán học khó khăn dựa trên một thuật toán băm mật mã SHA256. Giải pháp cho vấn đề này được gọi là bằng chứng xử lý, bằng chứng này được kèm theo một block mới và đóng vai trò là bằng chứng cho thấy người đào đã có nỗ lực tính toán rất lớn. Sự cạnh tranh để giải quyết bài toán Bằng chứng xử lý nhằm kiếm phần thưởng và quyền ghi các giao dịch lên blockchain là cơ sở cho mô hình bảo mật của bitcoin.

Nguồn cung ứng tiền bitcoin được tạo ra thông qua việc đào, tương tự như ngân hàng trung ương phát hành đồng tiền mới bằng cách in tiền. Số lượng bitcoin được tạo mới mà một thợ đào có thể thêm vào một block giảm dần sau 210 000 block. Số lượng tối đa bắt đầu ở mức 50 bitcoin mỗi block vào tháng 1 năm 2009 và tại thời điểm viết báo cáo là 12.5 bitcoin. Đến năm 2140 khi tất cả bitcoin được phát hành (20.99999998 triệu) thì sẽ không có bitcoin mới nào được phát hành.

#### 4.2 Đồng thuận phi tập trung

Không như các hệ thống thanh toán truyền thống phụ thuộc vào sự tin tưởng vào một bên thứ ba. Bitcoin không có một cơ quan quyền lực trung tâm nào nhưng mọi nút đầy đủ đều có một bản sao đầy đủ của một sổ cái công khai mà nút đó có thể tin tưởng như một bản ghi chính thức. Blockchain không được tạo ra bởi một cơ quan quyền lực nào mà được tổng hợp một cách độc lập bởi mọi nút trong mạng lưới.

Sự đồng thuận phi tập trung của Bitcoin nổi bật từ sự ảnh hưởng lẫn nhau của 4 tiến trình xảy ra độc lập tại các nút trên toàn mạng lưới:



- Mỗi nút bitcoin đầy đủ được độc lập xác minh các giao dịch dựa trên các tiêu chí.
- Các nút đào được độc lập tổng hợp các giao dịch và thêm vào những block mới, kết hợp với việc tính toán được thể hiện qua thuật toán bằng chứng xử lý.
- Tất cả các nút được độc lập xác thực tất cả các block mới và tổng hợp vào một chuỗi.
- Mỗi nút được độc lập chọn một chuỗi, chuỗi có tính toán được tích lũy nhiều nhất được thể hiện qua bằng chứng công việc.

#### 4.3 Độc lập xác thực các giao dịch

Sau khi các ứng dụng ví tạo ra các giao dịch bằng cách thu thập các UTXO, cung cấp các kịch bản mở khóa thích hợp và sau đó xây dựng các input mới để gán cho người sở hữu mới. Khi đó các giao dịch mới được gửi đến các nút lân cận trong mạng Bitcoin để được lan truyền khắp mạng lưới Bitcoin.

Khi một nút nhận được một giao dịch, trước khi chuyển tiếp giao dịch đó cho các nút lân cận của nó thì nó sẽ phải xác minh giao dịch đó trước tiên. Điều này đảm bảo rằng chỉ có các giao dịch hợp lệ mới có thể được truyền qua mạng và những giao dịch không hợp lệ sẽ bị loại khỏi mạng lưới ở ngay nút đầu tiên.

#### 4.4 Tổng hợp giao dịch vào các block

Sau khi xác thực các giao dịch, một nút Bitcoin sẽ thêm chúng vào vùng nhớ (*mempool*) hoặc vùng giao dịch (*transactions pool*), đây là nơi các giao dịch chờ cho đến khi chúng được thêm vào block tiếp theo. Các thợ đào sẽ được tự do lựa chọn các giao dịch để thêm vào block tiếp theo (thường là dựa theo mức phí giao dịch sẽ trả cho các thợ đào). Các block này được gọi là block ứng cử (*candidate block*).

Trong khi các thợ đào liên tục lắng nghe các giao dịch được lan truyền trong mạng lưới trong khi đang cố gắng đào một block mới và cũng lắng nghe các block được phát hiện. Ngay khi các thợ đào nhận được một block đã được xác thực, block này chứa các giao dịch từ 10 phút trước thì tức là cuộc cạnh tranh để giành lấy phần thưởng của 10 phút trước đã kết thúc và đã có một người chiến thắng.

Khi nhận được một block đã xác thực mới thì các thợ đào sẽ so sánh các giao dịch trong block đó với các giao dịch đang nằm trong vùng nhớ, các giao dịch trùng nhau sẽ bị loại bỏ đi. Các giao dịch còn lại sẽ được đưa vào một block rỗng cùng với trường “previousblockhash” và sẽ được đào ngay lập tức. Các block ứng cử chưa

phải là các block hợp lệ vì nó chưa có một bằng chứng xử lý hợp lệ. Chỉ khi lời giải cho thuật toán bằng chứng xử lý được tìm ra thì mới được xem là hợp lệ.

#### 4.5 Giao dịch coinbase

Trong bất kỳ một block nào, giao dịch đầu tiên được gọi là giao dịch coinbase. Giao dịch này được tạo ra và thêm vào bởi các thợ đào. Để xây dựng giao dịch coinbase, các thợ đào phải tính tổng số phí và giá trị phần thưởng cho block mới. Phần thưởng được tính theo chiều cao của block, bắt đầu là 50 bitcoin và giảm một nửa cứ sau mỗi 210.000 block. Khi tính được tổng số phí và phần thưởng và gửi nó lại vào địa chỉ bitcoin của chính thợ đào. Các giao dịch này không có input mà chỉ có duy nhất một output. Và sau 100 block được tạo ra ngay sau giao dịch chứa coinbase đó thì số bitcoin trong đó mới có thể sử dụng được.

#### 4.6 Đào và thuật toán bằng chứng xử lý

Đào là một quá trình dùng hàm băm (ở đây là hàm băm SHA256) để băm toàn bộ block lặp đi lặp lại, thay đổi các tham số sao cho phù hợp, cho đến khi kết quả băm phù hợp với một mục tiêu cụ thể. Kết quả của hàm băm là không thể xác định trước. Cách duy nhất để tìm ra một kết quả phù hợp với một mục tiêu cụ thể là phải thử đi thử lại, điều chỉnh các tham số cho đến khi tìm được một giá trị phù hợp xuất hiện ngẫu nhiên.

Có một giá trị trong block được gọi là nonce là một số nguyên có độ lớn  $2^{32}$ . Nonce được dùng để thay đổi giá trị input của hàm băm. Bằng chứng xử lý phải tạo ra một mã băm nhỏ hơn chỉ tiêu cho trước. Chỉ tiêu càng cao thì càng dễ và ngược lại.

Dưới đây là một địa chỉ của một block hợp lệ. Block này đã tìm ra lời giải mà mạng lưới Bitcoin đã đặt ra, nhỏ hơn một giá trị cho trước. Ở đây là nhỏ hơn một số có 18 chữ số 0 nằm ở đầu.

000000000000000000000000027f88013f67811ecbfc413081b76d0d0b5a7bebf05af

#### 4.7 Độ khó

Do Bitcoin hoàn toàn mở do đó ai cũng có thể tham gia vào hoặc rời đi cũng như các thợ đào liên tục nâng cấp phần cứng làm khả năng tìm được block mới nhanh hơn 10 phút. Nên mạng lưới Bitcoin có thể điều chỉnh lại độ khó sao chỉ có được một chỉ tiêu phù hợp, đảm bảo các thợ đào sẽ tìm lời giải trong vòng 10 phút.

Việc đặt lại chỉ tiêu diễn ra độc lập và tự động trên tất cả các nút. Cứ sau 2016 block thì tất cả các nút sẽ đặt lại chỉ tiêu bằng chứng xử lý. Nếu mạng lưới tìm ra các block nhanh hơn 10 phút thì độ khó sẽ tăng lên và ngược lại.

Công thức để tính toán lại độ khó như sau: Chỉ tiêu mới = Chỉ tiêu cũ \* (Thời gian thực đào 2016 block sau cùng / 2016 phút)

Độ khó của mạng lưới hoàn toàn động lập so với giá trị hay số lượng các giao dịch. Độ khó tăng lên là do các thợ đào và các máy đào công suất lớn ngày càng tham gia nhiều vào để cạnh tranh phần thưởng.

#### 4.8 Đào block thành công

Sau khi bất kì một thợ nào tìm ra lời giải phù hợp với chỉ tiêu thì họ sẽ ngay lập tức truyền block đó cho các nút lân cận. Các nút đó xác nhận và thêm vào blockchain của mình, sau đó tiếp tục một cuộc đua mới.

#### 4.9 Xác thực một block mới

Khi block mới được tìm ra nó sẽ được chuyển qua mạng lưới, mỗi nút thực hiện một loạt các phép thử để xác thực nó trước khi truyền tới các nút khác. Bởi vì các nút này xác thực hoàn toàn độc lập và có cùng một quy tắc giống nhau, nên bất cứ sự gian lận nào xảy ra điều không được chấp nhận trong mạng lưới Bitcoin. Điều này đảm bảo chỉ có các block hợp lệ mới được truyền qua mạng lưới. Đảm bảo rằng các block của các thợ đào trung thực sẽ được thêm vào blockchain, các thợ đào gian lận chỉ tốn kém chi phí mà không có bất cứ thu hoạch gì.

#### 4.10 Lựa chọn các chuỗi block

Khi một nút nhận được và đã xác nhận một block mới, nó sẽ tìm kiếm “previousblockhash” trong blockchain và thêm nó vào blockchain hiện tại. Một blockchain thông thường thường có ba nhánh:

- Nhánh chính, là nhánh dài nhất, chứa nhiều sức mạnh tính toán nhất.
- Các nhánh thứ cấp, là các nhánh rẽ ra từ nhánh chính.
- Và một tập các block mồ côi, các block này là hợp lệ nhưng không trở tới bất kì block nào nằm trong blockchain.

Chuỗi chính luôn là chuỗi hợp lệ, có nhiều block nhất trừ khi có hai nhánh dài bằng nhau. Các block cũng sẽ có các block anh em, các block này cũng hợp lệ nhưng không nằm trên nhánh chính. Chúng được lưu lại để có thể được tham chiếu đến trong tương lai trong trường hợp nhánh thứ cấp trở nên dài hơn nhánh chính.

Khi một nhánh thứ cấp trở nên dài hơn nhánh chính thì nhánh thứ cấp sẽ được chọn làm nhánh chính do chứa nhiều bằng chứng công việc hơn và nhánh chính lúc này sẽ trở thành nhánh thứ cấp.

Nếu một block hợp lệ được nhận mà không trở tới bất kì block cha nào thì sẽ được xem như là một block mồ côi. Block này sẽ được giữ lại và chờ cho tới khi cha của nó được truyền đến thì sẽ tiến hành thêm vào blockchain phù hợp.

Bằng cách chọn chuỗi dài nhất là chuỗi hợp lệ, nên các mâu thuẫn sẽ được giải quyết khi có một block mới được tạo ra và được thêm vào một chuỗi và khiến chuỗi đó dài hơn. Việc chọn chuỗi nào để thêm block vào, cho phép các thợ đào “bỏ phiếu” vào chuỗi mà mình tin tưởng. Do đó, quyền kiểm soát được đặt hoàn toàn vào tay người dùng.

#### 4.11 Phân nhánh

Do blockchain là một cấu trúc dữ liệu phi tập trung, các bản sao của nó không phải lúc nào cũng nhất quán. Các block có thể đến các nút khác nhau vào các thời điểm khác nhau, khiến cho các nút đó có những cái nhìn khác nhau về blockchain. Để giải quyết vấn đề này các block cố gắng mở rộng blockchain dựa trên nhánh có nhiều block nhất. Khi mà các nút luôn chọn chuỗi dài nhất làm chuỗi chính thì mạng bitcoin luôn ở trạng thái nhất quán.

Một sự không nhất quán tạm thời diễn ra khi mạng lưới có một đợt phân nhánh. Sự phân nhánh diễn ra bất cứ khi nào có hai thợ đào gần như tìm ra và lan truyền block ứng cử gần như đồng thời. Những nút nhận được một trong hai block đến trước sẽ thêm block đó vào nhánh chính và block đến sau vào nhánh thứ cấp.

Sau khi có hai nhánh bằng nhau đang tồn tại song song thì các thợ đào sẽ tiếp tục tìm lời giải cho các block tiếp theo dựa trên nhánh chính mà các thợ đào đã chọn từ block mà họ nhận được trước. Nếu một thợ đào tìm ra một block mới và lan truyền nó đi trong mạng lưới thì nhánh mà block đó chọn làm nhánh chính sẽ trở thành nhánh chính và nhánh còn lại sẽ trở thành nhánh phụ. Từ đó vấn đề phân nhánh tạm thời sẽ được giải quyết.

#### 4.12 Mining Pool

Khi mà độ khó ngày càng trở cao khiến cho các thợ đào riêng lẻ gần như không thể nào đào được. Cho nên các thợ đào sẽ tập hợp lại để tạo thành các *mining pool*. Tập hợp toàn bộ sức mạnh tính toán của những người tham gia vào và chia nhau các công việc để tìm ra một lời giải phù hợp.

Để mining pool biết được các thợ đào đã bỏ ra bao nhiêu sức mạnh tính toán để tìm ra đáp án phù hợp. Các mining pool sẽ đặt ra một chỉ tiêu thấp hơn nhiều so với chỉ tiêu mà mạng bitcoin đặt ra, để các thợ đào dù lớn hay nhỏ đều có thể kiếm được phần thưởng cho mình vì đã đóng góp một phần sức mạnh vào cho trang trại. Các thợ đào sẽ phải tìm ra các đáp án sao cho nhỏ hơn chỉ tiêu mà mining pool đặt ra.

Khi đó thì các thợ đào sẽ được xem như là đã đóng góp sức mạnh tính toán của mình vào mining pool đó. Và có thể một trong các giá trị đó là lời giải cho block tiếp theo.

Các mining pool thường sẽ chạy một hoặc nhiều nút đầy đủ cho phép các thợ đào chỉ cần kết nối tới các mining pool thông qua các giao thức riêng của mining pool, từ đó các thợ đào sẽ lấy các giao dịch về và thêm giao dịch coinbase gửi tới địa chỉ của mining pool và tiến hành quá trình đào.

Khi một thợ đào tìm được một lời giải phù hợp, thợ đào đó sẽ gửi giá trị (ở đây là nonce) về lại cho mining pool và mining pool sẽ ngay lập tức lan truyền block đó đi tới các nút lân cận, kết thúc quá trình chạy đua để bắt đầu một quá trình mới ngay sau đó.

Có hai loại mining pool: được quản lý và phi tập trung. Các loại mining pool được quản lý bởi một cá nhân hay tổ chức, loại này có ưu thế là các thợ đào chỉ cần chạy một nút đào không cần phải tải toàn bộ blockchain về. Nhưng bất lợi gây ra cho mạng lưới là tạo nên sự tập trung hóa, điều mà ban đầu Bitcoin muốn giải quyết. Loại thứ hai là mining pool phi tập trung, tạo ra một mạng lưới ngang hàng tương tự như mạng bitcoin nhưng nhỏ hơn nhưng sẽ phải yêu cầu các thợ đào phải chạy một nút đầy đủ.

#### 4.13 Tấn công đồng thuận

Thường được gọi là tấn công 51%, quá trình tấn công này xảy ra khi một nhóm hoặc thợ đào chiếm phần lớn sức mạnh của mạng tính toán của toàn mạng lưới. Khi đó họ có thể thay đổi lịch sử blockchain, đưa các giao dịch vào danh sách đen, có thể tự tăng tiền của bản thân. Không nhất thiết các thợ đào phải đạt trên 51% thì mới có thể đạt được quá trình này. Chỉ cần một nhóm thợ đào chiếm ưu thế hơn phần còn lại thì cũng có thể kiểm soát mạng lưới, nhưng 51% thì chắc chắn sự tấn công sẽ được thành công hoàn toàn.

Name	Symbol	Market Cap	Algorithm	Hash Rate	1h Attack Cost	NiceHash-able
Bitcoin	BTC	\$115.79 B	SHA-256	37,276 PH/s	\$473,207	1%
Ethereum	ETH	\$47.96 B	Ethash	240 TH/s	\$331,622	4%
Bitcoin Cash	BCH	\$12.64 B	SHA-256	5,391 PH/s	\$68,431	9%
Litecoin	LTC	\$4.62 B	Scrypt	299 TH/s	\$51,336	6%
Monero	XMR	\$2.22 B	CryptoNightV7	477 MH/s	\$17,150	14%
Dash	DASH	\$1.89 B	X11	2 PH/s	\$9,041	35%
Ethereum Classic	ETC	\$1.87 B	Ethash	13 TH/s	\$17,475	82%
Zcash	ZEC	\$723.73 M	Equihash	618 MH/s	\$45,424	8%
Bytecoin	BCN	\$552.57 M	CryptoNight	313 MH/s	\$705	155%
Bitcoin Gold	BTG	\$523.62 M	Equihash	18 MH/s	\$1,341	272%
Dogecoin	DOGE	\$299.92 M	Scrypt	263 TH/s	\$45,141	7%

Hình 3-2: Danh sách tỉ lệ các blockchain có thể bị tấn công.

Như ta thấy trên bảng trên, Bitcoin và Ethereum là hai blockchain có tỉ lệ tấn công thấp nhất. Như khi ta muốn tấn công Bytecoin trong vòng 1 giờ thì ta chỉ cần bỏ ra 705 đô la, số tiền này sẽ trả cho các mining pool để thực hiện tấn công.

#### 4.14 Phân nhánh cứng

Phân nhánh cứng diễn ra khi các quy tắc đồng thuận thay đổi, thường là do cập nhật lên các phiên bản mới nhưng một số nút vẫn ở phiên bản cũ, phiên bản mới không tương thích với phiên bản cũ. Việc này dẫn đến các nút trong mạng lưới không thể liên lạc được với nhau và sau đó tự ngắt kết nối. Việc này dẫn tới sự phân nhánh vĩnh viễn, chia một blockchain thành hai hoặc nhiều blockchain khác nhau cùng tồn tại, phát triển và cạnh tranh lẫn nhau.

Khi phân nhánh cứng diễn ra, các người dùng trong mạng lưới có quyền lựa chọn mình sẽ ở lại blockchain nào. Nếu blockchain nhận được nhiều người lựa chọn hơn từ phía người dùng sẽ có được sức mạnh tính toán lớn hơn và blockchain còn lại sẽ ít hơn. Và do đó, thời gian trung bình để tạo ra một block mới của cả hai blockchain sẽ bị giảm đi, gây ra tình trạng quá tải trên toàn hệ thống. Tuy nhiên, blockchain nhận được nhiều sự đồng thuận của người dùng sẽ xác thực nhanh hơn blockchain còn lại. Và phải sau 2016 block thì mới tốc độ tạo ra block mới mới có thể được cân bằng trở lại.

#### 4.15 Phân nhánh mềm

Phân nhánh mềm là quá trình cập nhật lên phiên bản mới nhưng vẫn giữ độ tương thích với các phiên bản cũ. Các giao dịch được tạo ra trên các blockchain phiên bản cũ sẽ vẫn được chấp nhận và xử lý bởi tất cả các nút trong hệ thống, nhưng các giao dịch của blockchain phiên bản mới thì sẽ không được chấp nhận và xử lý tại một số nút chạy phiên bản cũ hoặc có thể được chấp nhận nhưng sẽ bỏ qua các tính năng có trong blockchain bản mới.

Việc phân nhánh mềm không gây ra phân nhánh vĩnh viễn làm tách blockchain thành các blockchain khác nhau. Tuy nhiên, việc này sẽ rất khó để phát triển và dễ gây ra lỗi khi phải vừa tính toán sao cho phù hợp với phiên bản cũ. Thêm nữa là khi cập nhật lên bản mới thì sẽ không thể nào quay ngược lại được nữa, tức là khi có lỗi xảy ra, thì các nhà phát triển Bitcoin phải lập tức tạo ra một phân nhánh mềm hoặc phân nhánh cứng khác để sửa lỗi đó.

### III.5 An toàn bitcoin

Trên một mạng lưới thanh toán truyền thống, để mở một tài khoản tín dụng bạn cần phải cung cấp các thông tin cá nhân cho ngân hàng. Tất cả thông tin tập chung vào một chỗ có thể dễ dàng bị tấn công bởi tin tặc hoặc ngân hàng làm mất tiền. Và thông tin cá nhân có thể bị bán cho một bên thứ ba khác.

Trong Bitcoin, một giao dịch chỉ cho phép một giá trị cụ thể đến từ một người nhận cụ thể và không thể giả mạo hay sửa đổi. Nó cũng không tiết lộ bất kì thông tin cá nhân nào, chẳng hạn như danh tính các bên. Do đó mạng lưới Bitcoin không cần phải được mã hóa hoặc bảo vệ khỏi bị nghe trộm. Có thể phát tán các giao dịch qua các môi trường không an toàn như Wi-Fi hoặc Bluetooth mà không mất đi sự an toàn.

Do toàn bộ quyền kiểm soát đã được trao vào tay người dùng, nên người dùng phải tự có trách nhiệm đảm bảo các khóa luôn được giữ an toàn. Dưới đây là một số điều cần lưu ý để tránh bị đánh cắp hoặc thất lạc các khóa dẫn tới mất bitcoin vĩnh viễn:

- Các thiết bị máy tính, điện thoại thông minh thường kết nối với môi trường không an toàn, chứa nhiều phần mềm nguy hiểm, dễ dẫn tới bị đánh cắp.
- Việc đặt mật khẩu hoặc sao lưu các khóa quá phức tạp có thể dẫn tới mất bitcoin vĩnh viễn do không nhớ.
- Không nên giữ quá nhiều bitcoin trong cùng một ví, nên chia ra để lưu trong các ví một lượng nhỏ và không lưu quá nhiều bitcoin trên các ví online. Nên trữ bitcoin trong các ví offline.

- Nếu có quá nhiều bitcoin thì nên chia sẻ bí mật này với những người tin cậy hoặc luật sư để phòng trường hợp tai nạn bất ngờ hoặc qua đời.



# Chương 4. GIỚI THIỆU ETHEREUM

## I. ETHEREUM LÀ GÌ?

Do những giới hạn chế của Bitcoin, một lập trình viên tên là Vitalik Buterin đã tạo ra một nền tảng mới vào năm 2013, nền tảng đó cho phép có thể phát triển các ứng dụng chạy trên một nền tảng có những khả năng của Bitcoin nhưng tự do và có khả năng mở rộng hơn được gọi là Ethereum. Vào cuối tháng 7 năm 2015, Ethereum chạy chính thức trên toàn cầu.

Ethereum được xem như là một “Máy tính toàn cầu” phi tập trung. Chính xác hơn, Ethereum là một mã nguồn mở, một cơ sở hạ tầng điện toán phi tập trung cho phép chạy những chương trình được gọi là smart contract. Nó sử dụng blockchain để đồng bộ hóa và lưu trữ các trạng thái cùng với một đồng tiền được gọi là *ether* để đo lường và ràng buộc các tài nguyên mà smart contract của thể sử dụng trên Ethereum.

Các phép tính được thực hiện trên Ethereum được tính phí bằng một đơn vị gọi là gas và được giới hạn bởi gas limit. Gas sẽ do thị trường quy định dựa vào nhu cầu của thị trường. Tổng số phí sẽ được tính bằng công thức:  $\text{Phí} = (\text{Gas} * \text{Các phép tính})$ . Và tổng số phí này không được vượt quá gas limit nếu không giao dịch sẽ xem như thất bại và toàn bộ số gas sẽ không được hoàn trả lại. Ngược lại, nếu giao dịch hoàn thành thì gas thừa sẽ được trả lại vào tài khoản của người dùng.

Mục đích chính của Ethereum không phải là một mạng lưới thanh toán. Đồng tiền ether được dùng để thực thi các smart contract chạy trên Ethereum. Nhưng người dùng cũng có thể giao dịch hoặc mua bán các đồng tiền này và đổi lấy bitcoin hoặc những đồng khác.

## II. SỰ KHÁC NHAU GIỮA BITCOIN VÀ ETHEREUM

Bảng dưới để thể hiện một số phần khác biệt chính giữa Ethereum và Bitcoin

Ethereum	Bitcoin
<b>Smart Contract Platform</b> <ul style="list-style-type: none"> <li>Có một ngôn ngữ kịch bản turing complete</li> <li>Các đoạn mã có địa chỉ riêng của mình.</li> </ul>	<b>Scripting</b> <ul style="list-style-type: none"> <li>Có một ngôn ngữ stack đơn giản, không phải là turing complete</li> <li>Các đoạn mã nằm bên trong các giao dịch</li> </ul>
Các giao dịch dựa trên Accounts	Các giao dịch dựa trên UTXO
Thời gian tạo khối mới là xấp xỉ 15 giây	Thời gian tạo khối mới là xấp xỉ 10 phút
Sử dụng hàm băm ethash trong quá trình mining. Vẫn chưa có các thiết bị đào chuyên dụng nên chưa lo lắng về vấn đề tập trung hóa	Sử dụng hàm băm SHA256 trong quá trình mining. Có thể sử dụng các thiết bị đào chuyên dụng và gây ra tập trung hóa trong Bitcoin.
Nguồn cung tiền của không giới hạn	Giới hạn ở mức 21 triệu bitcoin
Cứ mỗi 3 ether được tạo ra xấp xỉ mỗi 15 giây	Cứ mỗi 12.5 bitcoin được tạo ra xấp xỉ mỗi 10 phút
Được quy định bởi gas limit, các giao dịch phải nằm trong một gas limit được quy định trước.	Kích thước mỗi block giới hạn 1MB

### II.1 Smart Contract và Scripting

Smart contract là các chương trình và sẽ được biên dịch và triển khai trên Ethereum. Khi bạn thực thi một chức năng được viết trong smart contract thì nó tất cả các nút trên hệ thống sẽ đồng thời cùng chạy và cho ra cùng một kết quả như nhau để đạt được một sự đồng thuận trên hệ thống.

Các giao dịch đều có tính nguyên tử (atomic), khi một giao dịch thành công và không hề có bất cứ lỗi nào xảy ra thì nó mới được chấp nhận và sẽ thay đổi trạng thái của contract. Ngược lại, nếu có bất kì lỗi nào xảy ra thì toàn bộ sự thay đổi sẽ bị trả lại thời điểm trước khi sự thay đổi xảy ra và xem như chưa có gì xảy ra. Tuy nhiên, các giao dịch thất bại vẫn sẽ được lưu vào blockchain và trừ đi tất cả các gas của người dùng gửi kèm theo giao dịch.

Mặc dù smart contract không thể bị thay đổi nhưng có thể “xóa” nó. Khi một contract bị xóa thì các trạng thái của nó sẽ bị xóa trong tương lai, nhưng các lịch sử giao dịch sẽ vẫn được giữ lại trong blockchain do blockchain là bất biến.

Smart contract có thể được viết bởi nhiều ngôn ngữ khác nhau, các ngôn ngữ này sẽ biên dịch ra cùng các mã để có thể chạy trên Ethereum, một số ngôn ngữ phổ biến dùng để viết smart contract là: Solidity, LLL, Vyper, ... Dưới đây là một đoạn mã đơn giản được viết bằng Solidity.

```
1. pragma solidity ^0.4.22;
2.
3. contract Hello {
4.     function getName() public view
5.         returns (string name)
6.     {
7.         name = "linh";
8.     }
9. }
```

Solidity là một ngôn ngữ được gọi là *Contract Oriented Programming (COP)* – Lập trình hướng hợp đồng. Ngôn ngữ này khá giống với một trong những ngôn ngữ phổ biến nhất hiện nay là JavaScript để giúp các lập trình dễ dàng học và tạo ứng dụng dựa trên ngôn ngữ này. Để hiểu rõ ta sẽ xem xét qua đoạn mã mẫu ở trên. Trong đoạn trên dòng 1 dùng để chỉ phiên bản sẽ sử dụng lớn hơn hoặc bằng phiên bản ^0.4.22. Dòng thứ 3 dùng để khai báo contract tên là Hello. Dòng 4 là khai báo hàm tên là getName, public để chỉ hàm này sẽ có thể được gọi từ cả bên trong và bên ngoài, view để chỉ hàm này chỉ trả về dữ liệu chứ không thay đổi hay đọc bất cứ biến nào trong contract, dòng returns (string name) để chỉ hàm này sẽ trả về một biến tên là name có kiểu dữ liệu là string.

Nếu muốn đọc hay ghi dữ liệu từ một contract bạn phải chạy một nút Ethereum đầy đủ, nhưng không phải ai cũng có thể chạy được nút đầy đủ do chi phí về phần cứng và băng thông. Do đó, chúng ta lại phải nhờ vào các nhà cung cấp thứ ba hỗ trợ việc này bằng cách họ sẽ chạy một loạt các máy chủ hoạt động như một nút đầy đủ, trong đó Infura là một trong những nhà cung cấp hàng đầu. Và ta sẽ kết nối tới các nút đó thông qua thư viện web3js được viết bằng JavaScript.

Không như Ethereum hỗ trợ viết một ứng dụng hoàn chỉnh dựa vào smart contract. Bitcoin chỉ có một ngôn ngữ kịch bản đơn giản được gọi là scripting. Scripting không được hỗ trợ để viết các ứng dụng phức tạp trên đó, mục đích chỉ là để khóa và mở khóa các giao dịch. Dưới đây là đoạn mã minh họa cho việc cộng hai số và so sánh với một số khác được viết bằng scripting của Bitcoin:

```
1. 2 7 OP_ADD 3 OP_SUB 1 OP_ADD 7 OP_EQUAL
```

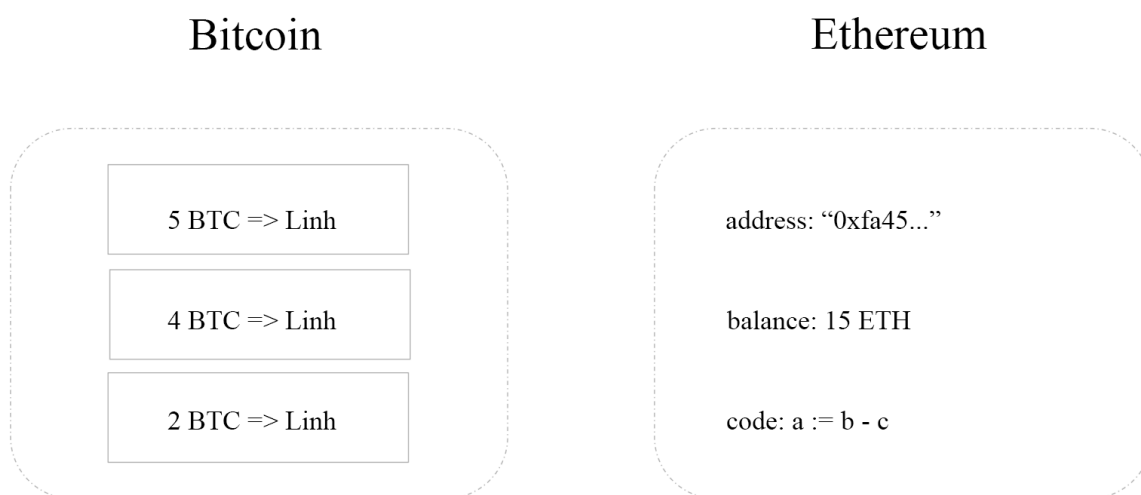
Scripting của Bitcoin được thiết kế rất đơn giản, trong đoạn trên dùng để mô tả cách cộng hai số và so sánh với số thứ ba. Đầu tiên Bitcoin sẽ đưa đoạn mã trên vào một ngăn xếp (*stack*). Các đoạn mã được thực thi từ trái sang phải:

- 2 7 OP\_ADD: lấy hai số 2 và 7 cộng là và cho ra kết quả là 9, sau đó đẩy lại vào stack.
- 9 3 OP\_SUB: lấy 9 trừ 3 bằng 6 sau đó đẩy lại vào stack.
- 6 1 OP\_ADD: lấy 6 cộng 1 bằng 7 và đẩy lại vào stack.
- 7 7 OP\_EQUAL: so sánh hai số 7 và 7 và kết quả trả về là TRUE do 7 bằng 7. Ngược lại sẽ trả về FALSE.

Qua đó ta thấy được Ethereum hoàn toàn chiếm ưu thế về mặt này. Thêm nữa, scripting của Bitcoin không hỗ trợ vòng lặp, không có địa chỉ riêng chỉ có thể được kèm theo các giao dịch và có kích thước giới hạn. Ngược lại, Smart Contract hỗ trợ đầy đủ như một ngôn ngữ lập trình bình thường. Để làm quen hơn, tuy nhiên việc hỗ trợ nhiều tính năng hơn đồng nghĩa với việc có khả năng có nhiều lỗi tiềm ẩn hơn.

## II.2 Accounts và UTXO

Đối với Bitcoin số dư được tính bằng tổng số các UTXO mà người đang sở hữu khóa bí mật tương ứng. Thay vào đó, Ethereum sử dụng một khái niệm hoàn toàn khác được gọi là Accounts, nó luôn theo dõi số dư tài khoản của người dùng và sẽ tiến hành cập nhật lại số dư đó khi trạng thái của hệ thống được cập nhật.

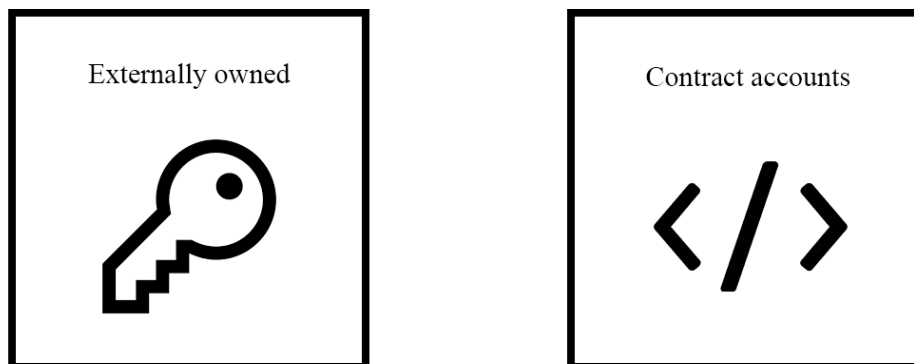


Hình 4-1: UTXO trong Bitcoin và Accounts trong Ethereum.

Trong Ethereum có hai loại accounts là

Người sở hữu bên ngoài (*Externally owned*): sẽ được kiểm soát thông qua khóa bí mật và không liên quan đến bất cứ đoạn mã nào.

Tài khoản hợp đồng (*Contract accounts*): được kiểm soát bởi các đoạn mã bên trong nó, sẽ chứa và thực thi các đoạn mã.



Hình 4-2: Externally owned và Contract accounts

### II.3 Thời gian tạo khối mới

Bitcoin tạo một block mới trong vòng xấp xỉ 10 phút và kích thước block khối xấp xỉ 1 MB do Bitcoin đã thiết lập một block chỉ có tối đa 1 MB, trong khi đó Ethereum tạo một block mới chỉ trong vòng xấp xỉ 15 giây nhanh hơn nhiều so với Bitcoin nhưng kích thước mỗi block lại thấp hơn nhiều. Kích thước block cao nhất từng được ghi nhận trong Ethereum là gần 34 KB. Ethereum giới hạn kích thước của một block thông qua gas limit. Toàn bộ chi phí của các phép tính trong một khối không được vượt quá *gas limit* này. Việc này đồng nghĩa với ít giao dịch được xử lý hơn nhưng việc xử lý lại được diễn ra nhanh hơn.

# Chương 5. ỨNG DỤNG GỌI VỐN TRÊN NỀN TẢNG ETHEREUM

## I. GIỚI THIỆU

Với sự phát triển của cuộc cách mạng công nghiệp 4.0 hiện nay, rất nhiều start-up ra được ra đời. Nhưng việc kêu gọi đầu tư từ các nhà đầu tư là một việc vô cùng khó khăn. Thay vào đó, các start-up sẽ kêu gọi vốn từ cộng đồng. Một trong những công ty nổi bật nhất trong lĩnh vực kêu gọi vốn là Kickstarter.

Kickstarter cho phép những nhà kinh doanh có khả năng phát triển sản phẩm trình bày dự án của mình, nhằm gọi vốn từ người dùng trên Kickstarter ở phạm vi toàn cầu. Khi một dự án được đưa lên Kickstarter để kêu gọi vốn, dự án bắt buộc phải xác định mức vốn đầu tư cần có và thời gian thực hiện chiến dịch gọi vốn cho dự án rõ. Một số điểm mạnh của Kickstarter:

- Có một cộng đồng người dùng lớn.
- Là ứng dụng tập trung nên có tốc độ xử lý cao.
- Nếu dự án không thành công, tiền ủng hộ sẽ được hoàn trả cho chủ đầu tư.
- Chủ dự án bắt buộc phải thực hiện dự án sau khi nhận tiền và trả lãi hoặc phần thưởng theo thỏa thuận ban đầu cho nhà đầu tư.

Tuy vậy, Kickstarter vẫn mang nhiều điểm bất cập:

- Nếu dự án thất bại người dùng sẽ không nhận được gì cả trong khi phải tốn rất nhiều tiền để marketing, làm video, chụp ảnh sản phẩm.
- Khi dự án thành công, Kickstarter thu 5% trên tổng số tiền huy động được, số tiền còn lại sẽ được chuyển cho chủ dự án.
- Thông tin về khách hàng, dự án điều được thu thập, mức tính phí khá cao có thể sẽ ảnh hưởng đến dự án. Thêm 3 – 5% các chi phí chuyển tiền, pháp lý khác.

Với những tính chất có được ở Blockchain - ở đây là smart contract - có thể giải quyết được vấn đề này. Với khả năng chuyển tiền trực tiếp một cách nhanh chóng, ẩn danh và chi phí thấp sẽ không làm ảnh hưởng đến tiến độ dự án mà vẫn đảm bảo tính riêng tư, bảo mật. Một khi smart contract được triển khai thì sẽ không thể thay đổi được, đảm bảo sự minh bạch trong quá trình gây quỹ.

Lý do mà tôi chọn Ethereum là do Ethereum là một nền tảng phi tập trung, đã đi vào hoạt động ổn định trong nhiều năm, được hỗ trợ bởi những công cụ lập trình tốt nhất hiện nay và không bị tấn công 51% như hầu hết blockchain khác, cũng như được kiểm tra, phát triển, nâng cấp qua nhiều phiên bản, đồng thời cho phép các ứng dụng chạy trên đó. Không như hầu hết các blockchain hiện nay không phải thực sự phi tập trung mà sẽ phụ thuộc vào một bên thứ ba. Và một phần quan trọng nữa là do Ethereum có một cộng đồng lớn hoạt động thường xuyên có thể giúp đỡ ngay khi có lỗi phát sinh.

Ethereum	ETH	\$47.96 B	Ethash	240 TH/s	\$331,622	4%
Bitcoin Cash	BCH	\$12.64 B	SHA-256	5,391 PH/s	\$68,431	9%
Litecoin	LTC	\$4.62 B	Scrypt	299 TH/s	\$51,336	6%
Monero	XMR	\$2.22 B	CryptoNightV7	477 MH/s	\$17,150	14%
Dash	DASH	\$1.89 B	X11	2 PH/s	\$9,041	35%
Ethereum Classic	ETC	\$1.87 B	Ethash	13 TH/s	\$17,475	82%

Hình 5-1: Danh sách các blockchain có khả năng bị tấn công 51%

Ta thấy ở đây có cả Ethereum và Ethereum Classic, cả hai đều hỗ trợ chạy smart contract như nhau, nhưng Ethereum Classic lại có tới 82% bị tấn công nên tôi quyết định chọn Ethereum làm nền tảng để triển khai dự án này.

## II. ỨNG DỤNG GỌI VỐN LIGHTHOUSE

Ở đề tài tốt nghiệp luận văn lần này, tôi đã phát triển một website hỗ trợ gọi vốn. Trang web này được gọi là Lighthouse. Lighthouse ra đời nhằm mục đích giải quyết những vấn đề còn tồn đọng trên các hệ thống gây quỹ truyền thống. Với các tính năng nổi bật sau:

- Mã nguồn bất biến: ứng dụng cho phép người dùng tạo ra các dự án gây quỹ trên smart contract với số tiền, thời gian gây quỹ được xác định trước và không thể thay đổi sau đó, khi mà dự án được triển khai lên blockchain.
- Không cần người trung gian: cho phép các bên liên quan giao dịch trực tiếp với nhau mà không cần thông qua một bên trung gian mà vẫn giữ được sự tin cậy với nhau.
- Xác thực danh tính: người dùng muốn tạo dự án thì họ phải tiến hành xác minh danh tính tương ứng với bộ khóa mà họ muốn sử dụng để tạo ra ứng dụng. Điều này một lần nữa hạn chế tình trạng spam và gian lận trên ứng dụng.

- Giao dịch ẩn danh: người dùng bình thường có thể giao dịch một cách hoàn toàn ẩn danh mà không cần phải lo lắng nhiều về thông tin các nhân sẽ bị lộ cho bất kì một bên nào.
- Giao dịch nhanh chóng: cũng giao dịch phi biên giới một cách nhanh chóng mà không cần phải đợi một thời gian khá lâu, có cả năng thất thoát, chi phí cao.
- Minh bạch: ứng dụng cũng ràng buộc ràng người gây quỹ chỉ có thể rút tiền ra khi mà thời gian gây quỹ kết thúc, nếu đến hết thời hạn mà không đạt được mục đích thì toàn bộ số tiền sẽ được trả lại cho người gây quỹ một cách tự động. Mọi giao dịch đều công khai, có thể được kiểm tra bởi bất kì ai trên mạng lưới.
- Tài nguyên phi tập trung: các hình ảnh, nội dung được lưu trữ trên một hệ thống lưu trữ tập tin phi tập trung, điều này cho phép các hình ảnh và nội dung này tồn tại mãi cùng với smart contract mà không bị bất cứ một ai có thể tác động đến. Không cần lo lắng về việc sao lưu dữ liệu, tận dụng tài nguyên sẵn có của các cá nhân trên toàn thế giới.
- Không cần mật khẩu: ứng dụng không cần phải thiết lập mật khẩu đăng nhập cũng như là tài khoản người dùng, mọi thứ đều được xác định thông qua ví mà người dùng đã cài đặt vào máy, việc này vừa thuận tiện, vừa làm cho người dùng phải có trách nhiệm tự quản lý các bộ khóa của mình một cách an toàn.
- Gần như không thể đánh cắp thông tin: Mọi nội dung đều được lưu trữ một cách phi tập trung nên đã hạn chế việc bị theo dõi đánh cắp thông tin người dùng. Và mật khẩu cũng không được lưu tập trung lại một chỗ cho nên việc đánh cắp thông tin gần như bất khả thi.
- Điện toán đám mây: để tận dụng điểm mạnh của các nền tảng phi tập trung và tập trung. Việc sử dụng Amazon Web Services EC2 để triển khai ứng dụng front-end để tăng trải nghiệm người.
- Hoàn toàn mở: do mã nguồn hoàn toàn công khai nên bất kì ai cũng có thể kiểm tra, đánh giá và thông báo với cộng đồng nên sẽ đảm bảo an toàn cho mọi người.
- HTTPS2 – Server Push: công nghệ HTTPS2 được sử dụng trên IPFS cho phép máy khách chỉ cần gửi một yêu cầu và máy chủ sẽ gửi đồng loạt nhiều tài



nguyên mà máy khác yêu cầu, làm tăng tốc độ chuyển tập tin và giảm băng thông.

Hiện tại ứng dụng sẽ không tính phí tạo dự án, dự án sẽ được tạo chỉ tốn một ít tiền của người dùng (số tiền này phụ thuộc vào thị trường và những tham số mà người dùng nhập vào), việc tốn chi phí khi tạo một dự án cũng sẽ hạn chế những dự án spam một cách tự động, khiến cho việc spam trở nên cực kì tốn kém.

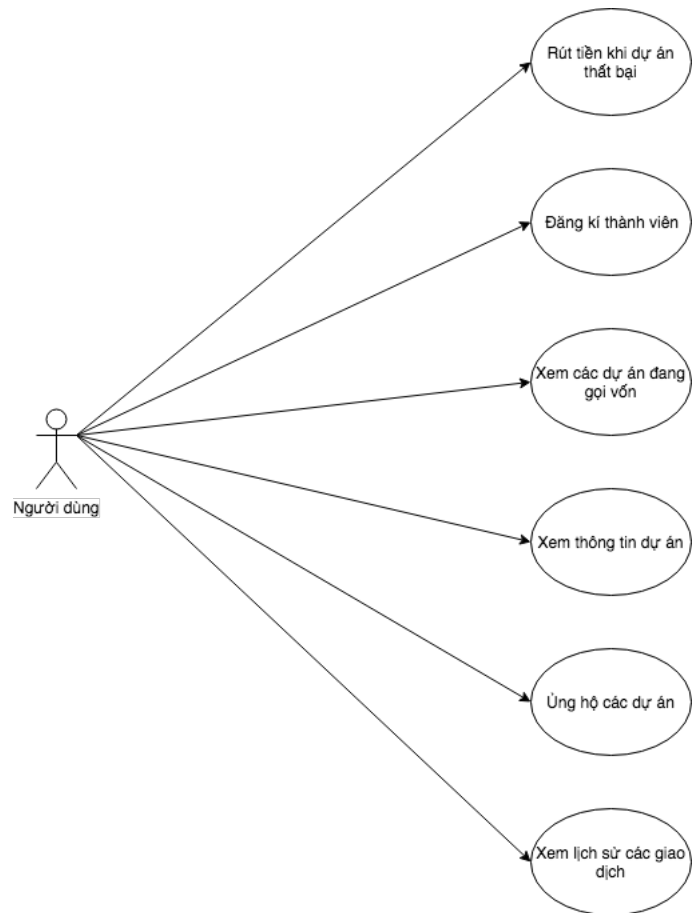
Ứng dụng này được gọi là Lighthouse (ngọn hải đăng) như một ngọn hải đăng soi sáng đường đi cho các thủy thủ để có thể đến đích an toàn. Ứng dụng này có tên gọi như thế cũng vì nó cũng làm những chức năng tương tự nhưng không phải đối với các thủy thủ mà là đối với các start-up, tổ chức từ thiện, làm cho dự án của họ có thể đạt được mục đích, tạo ra những sản phẩm tốt, giúp đỡ mọi người, làm thế giới phát triển hơn. Giúp các start-up, tổ chức thấy rõ đường đi (minh bạch) mà không gặp phải trở ngại ẩn đằng sau bóng tối (như các tổ chức truyền thống bạn không biết có gì sẽ xảy ra bên trong). Ngọn hải đăng luôn điều đặn cứ sau khoảng thời gian nhất định sẽ đi qua vị trí cũ lần nữa, tương tự như smart contract luôn được xử lý điều đặn mỗi 15 giây. Sau đây là những công nghệ được sử dụng để xây dựng nên một “Ngọn hải đăng”.

### III. VAI TRÒ

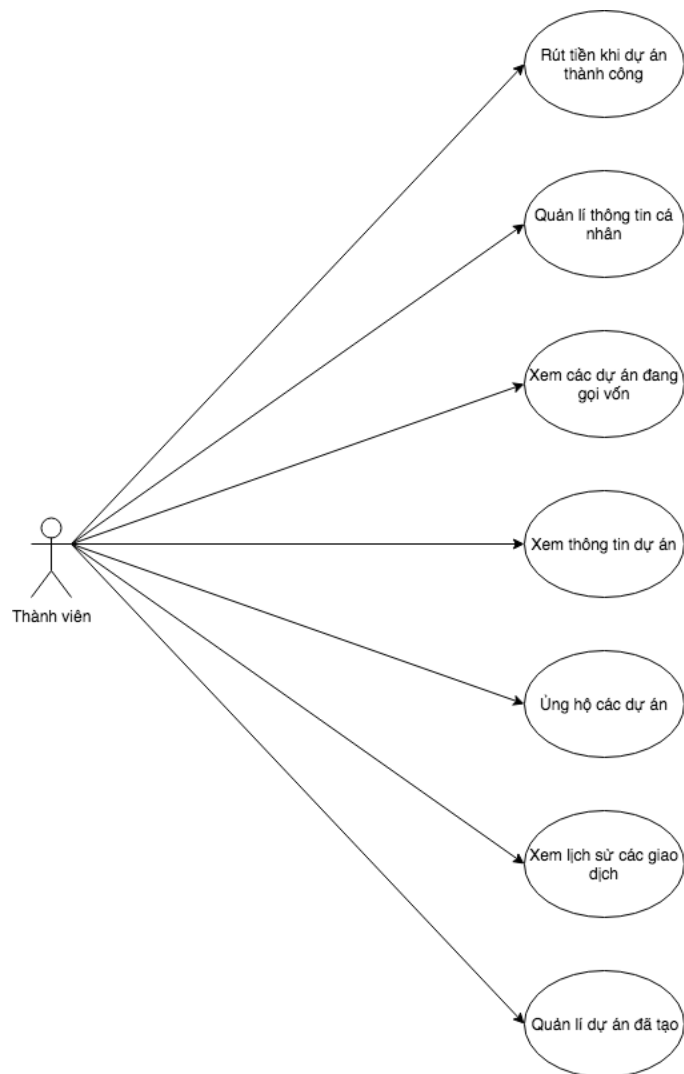
Trong website Lighthouse sẽ có các tác nhân sử dụng chính là:

- Người dùng: cho phép ủng hộ các dự án một cách ẩn danh, xem các thông tin liên quan đến dự án, đăng kí để trở thành thành viên để có thể tạo các dự án.
- Thành viên: cho phép ủng hộ các dự án khác nhưng sẽ không ẩn danh, xem các thông tin liên quan đến dự án của mình và của người khác, thay đổi thông tin dự án (cho phép một số thông tin được cho phép).
- Người triển khai ứng dụng Lighthouse: tương tự như người dùng bình thường.

Các vai trò sẽ được biểu diễn qua các sơ đồ sau:



Hình 5-2: Sơ đồ chức năng của người dùng



Hình 5-3: Sơ đồ chức năng của thành viên



Hình 5-4: Sơ đồ chức năng của người sở hữu Lighthouse

## IV. CÔNG NGHỆ SỬ DỤNG

Một ứng dụng phi tập trung không chỉ chạy trên smart contract mà nó còn phải được xây dựng từ nhiều công nghệ khác nhau. Trong đó:

- Smart contract: có vai trò như là database và cả server. Chịu chức năng lưu trữ và xử lý các logic của ứng dụng.
- Solidity: do smart contract có thể được viết bằng nhiều ngôn ngữ khác nhau nhưng tôi chọn Solidity vì nó là ngôn ngữ chính thức được phát triển bởi

Ethereum. Solidity khá giống với JavaScript nên sẽ dễ dùng và ít tốn thời gian hơn.

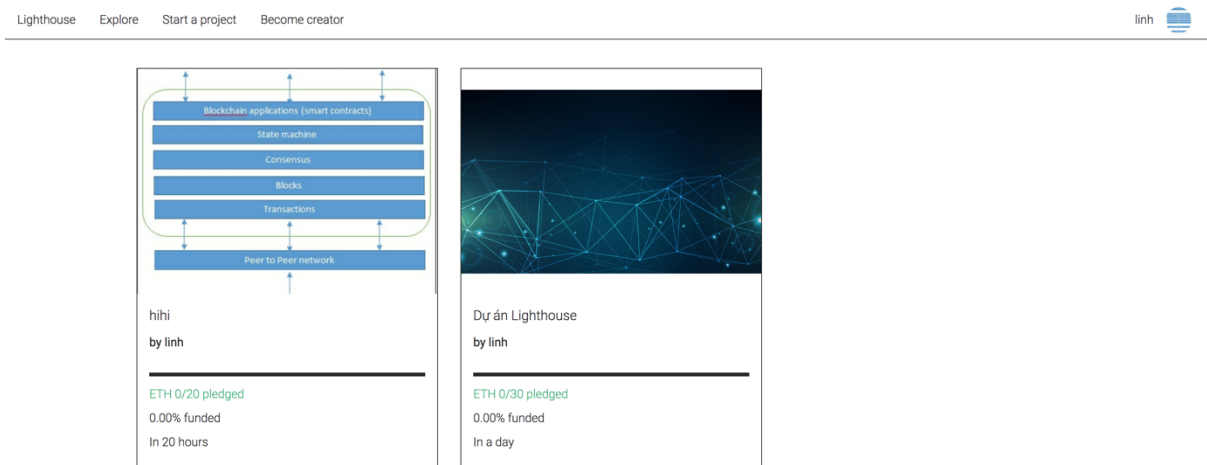
- React (Giao diện người dùng, Front-end): để người dùng tương tác, ta cần một giao diện để người dùng có thể tương tác tốt. React là một thư viện được viết bằng ngôn ngữ JavaScript dùng để xây dựng các ứng dụng front-end một cách nhanh chóng, mạnh mẽ, trải nghiệm người dùng tốt hơn. Các ứng dụng viết bằng React cho phép khi chuyển trang sẽ không cần phải tải lại toàn bộ trang mà chỉ tải lại những phần mà nó thay đổi, làm tăng tốc độ cũng như tài nguyên hệ thống của cả máy chủ và máy khách. React được Facebook phát triển và sử dụng và là thư viện hàng đầu hiện nay.
- Webpack: là một công cụ nén nhỏ tất cả các thư viện JavaScript, HTML, CSS lại thành một tập tin cho mỗi loại để có thể truyền tải nhanh hơn khi tải web, giúp tăng trải nghiệm người dùng, giảm băng thông.
- IPFS (Hệ thống lưu trữ tập tin liên hành tinh, InterPlanetary File System): Hệ thống IPFS cho phép lưu các tập tin lên một mạng lưới phi tập trung. Các tập tin này không thể xóa mà chỉ có thể được cập nhật. Khi tải lên hệ thống sẽ cắt nhỏ nội dung tập tin và lưu trữ ngẫu nhiên trên toàn mạng lưới, và do bị cắt nhỏ nên những máy tính lưu nó không thể biết được nội dung bên trong. Các tập tin cũng được sao chép ra nhiều lần và lưu tại nhiều nơi để hạn chế điểm chết. Một ưu điểm nữa là khi hai hay nhiều tập tin giống nhau được tải lên hệ thống thì nó vẫn chỉ lưu một cái và trả về ảnh tương tự cho người dùng. Do đó, giảm được rất nhiều chi phí về tài nguyên. Và do mạng lưới càng trở nên mạnh khi có nhiều người dùng tham gia, không như các hệ thống truyền thống càng nhiều người dùng sẽ gây nên tình trạng quá tải.
- Truffle: một tập hợp các công cụ dùng để triển khai, nâng cấp, kiểm thử, migrations với các phiên bản cũ, hỗ trợ blockchain ảo cho việc lập trình. Bên cạnh Truffle có nhiều công cụ khác, nhưng đây vẫn là công cụ đi đầu và được hỗ trợ nhiều nhất bởi cộng đồng.
- Webstorm: là một ứng dụng dùng để lập trình các ứng dụng web liên quan chủ yếu tới JavaScript, HTML, CSS, đồng thời Webstorm cũng hỗ trợ Solidity với các chức năng như gợi ý, syntax highlight, kiểm lỗi.
- GitHub: là một trang quản lý phiên bản phổ biến nhất hiện nay, cùng với các tính năng như một mạng xã hội dành cho lập trình viên. Github hoàn toàn miễn phí cho các dự án công khai. Do các smart contract khi được triển khai

lên Ethereum thì vẫn sẽ được xem hết nội dung bên trong kể cả các đoạn mã nên việc tạo dự án công khai trên GitHub không có vấn đề gì.

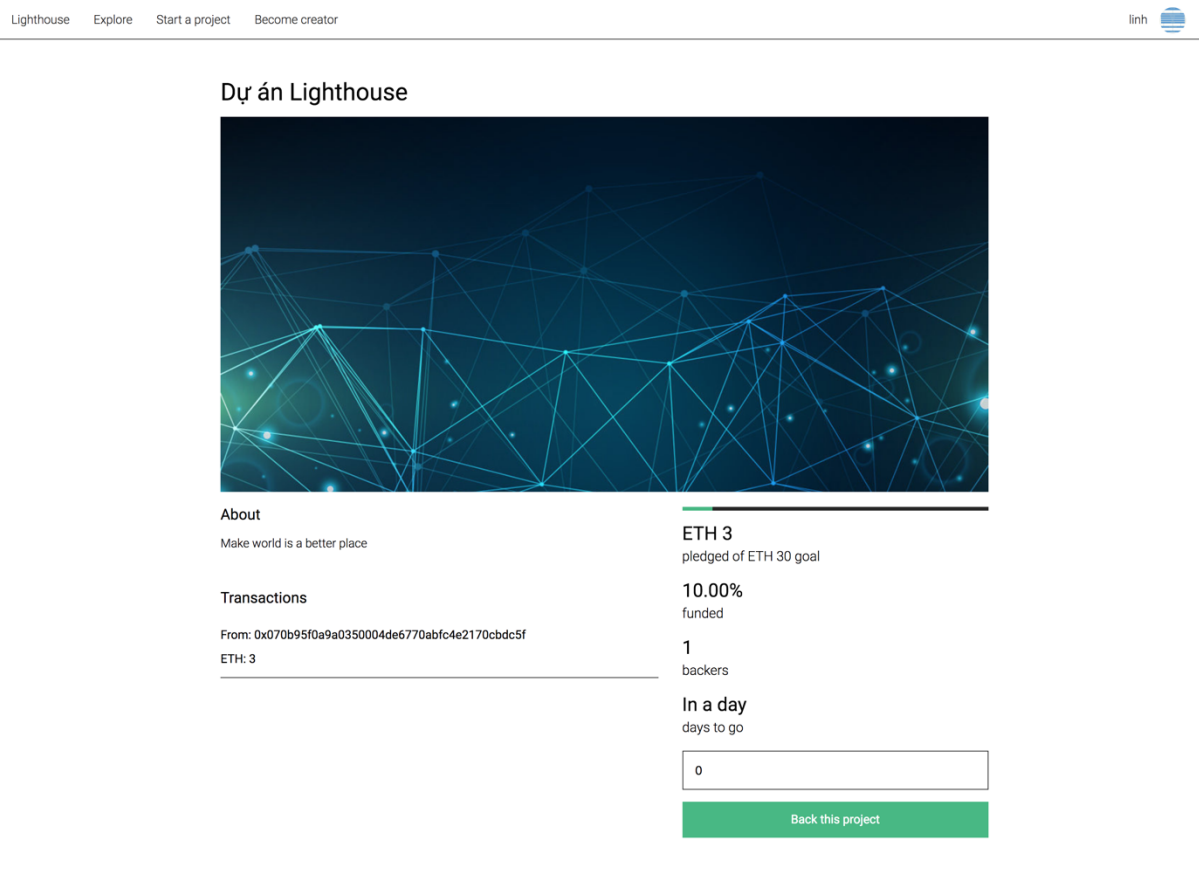
- AWS EC2 (Elastic Cloud): là một dịch vụ của Amazon Web Services cung cấp giải pháp điện toán đám mây cho người dùng. Cho phép tạo và quản lý một server dễ dàng, đặt tại nhiều khu vực khác nhau trên thế giới. Mặc dù smart contract là phi tập trung nhưng để có được một sự trải nghiệm tốt, nhanh chóng thì một nền tảng phi tập trung hiện nay chưa phù hợp cho phí đắt đỏ và chậm chạp do đó nên dự án tạm thời sẽ chạy trên một cơ sở hạ tầng tập trung.
- MetaMask: có thể xem MetaMask như là một cầu nối cho phép kết nối tới các contract nằm trên các nút của Ethereum thông qua trình duyệt mà không cần phải khởi chạy Ethereum. Đồng thời MetaMask cũng là một ứng dụng ví chạy trên trình duyệt, cho phép tạo, quản lý và ký lên các giao dịch trên Ethereum một cách dễ dàng và nhanh chóng.
- Nginx: là 1 máy chủ reverse proxy mã nguồn mở cho các giao thức HTTP, HTTPS, SMTP, POP3 và IMAP, cũng như là 1 máy chủ cân bằng tải (load balancer), HTTP cache và web.

## V. HÌNH ẢNH

Ứng dụng tập trung vào trải nghiệm người dùng hơn là giao diện người dùng nên giao diện được thiết kế một cách đơn giản nhất nhằm tăng tốc độ tải trang, giảm băng thông. Sau đây là một số hình ảnh:



Hình 5-5: Trang chủ của Lighthouse hiển thị các dự án



Hình 5-6: Trang hiển thị thông tin chi tiết của dự án

LighthouseExploreStart a projectBecome creator

linh

User Info

Transactions

My Projects

Username

linh

Email

linh@gmail.com

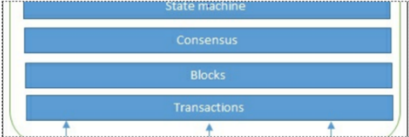
Address

HCM VN

Biography

Hahaha

Avatar



Update

Hình 5-7: Trang quản lý thông tin cá nhân

LighthouseExploreStart a projectBecome creator

linh

Enter your project name

Be careful!!! You can't change it after create.

Ex: Hello world

Describe what you'll be creating.

You can't also edit this later, too.

Ex: Make world is a better place

How many ETH you want to raise?

Be careful !!! You can't change it after create.

0

When is the project end?

Be careful !!! You can't change it after create.

0

Add thumbnail

Be careful !!! You can't change it after create.

Try dropping some files here, or click to select files to upload.

Create

Hình 5-8: Trang tạo dự án

Username

Ex: Linh The Human

Email

Ex: linrium@gmail.com

Address

Ex: 20

Biography

Ex: Make world is a better place

Avatar

Try dropping some files here, or click to select files to upload.

Create

Hình 5-9: Trang đăng kí thành viên



# Chương 6. TỔNG KẾT

## I. ƯU NHƯỢC ĐIỂM

Qua quá trình phát triển sản phẩm, dự án tuy đã hoàn thành tốt với những ưu điểm nổi bật nhưng cũng không tránh khỏi không ít khó khăn, dưới đây là những ưu nhược điểm đã đạt được.

Ưu điểm của ứng dụng:

- Giao diện đơn giản, tập trung vào trải nghiệm người dùng hơn là giao diện người dùng.
- Các giao dịch diễn ra một cách minh bạch, nhanh chóng do mọi giao dịch đều được công khai, việc ủng hộ một cách phi biên giới dễ dàng hơn (không bị tính mức phí cao khi sử dụng Visa hay Paypal).
- Mã nguồn bất biến và mở hoàn toàn làm cho trang web trở nên minh bạch, có thể được kiểm tra và báo cáo bởi cộng đồng. Điều mà các Kickstarter không có được, người dùng sẽ không biết bên trong là gì hoạt động như thế nào.
- Loại bỏ người trung gian cho phép hai bên (người dùng và người tạo dự án) giao dịch trực tiếp với nhau, giảm những chi phí khác phải trả cho bên thứ ba.
- Cho phép giao dịch ẩn danh, không tiết lộ bất kì thông tin gì của người dùng. Làm các giao dịch trở nên an toàn hơn trong môi trường không an toàn.
- Xác thực danh tính tin cậy hơn do người dùng phải bỏ ra một mức phí lớn để thêm thông tin vào blockchain nên sẽ hạn chế các tài khoản spam vì chi phí sẽ trở nên rất lớn.
- Không thể bị tắt hay đánh cắp thông tin do toàn bộ thông tin đều mở và ẩn danh nên việc đánh cắp gần như bất khả thi. Và không ai hiện tại có đủ khả năng tắt được ứng dụng.
- Không cần mật khẩu, không cần phải gõ lại mật khẩu mỗi lần đăng nhập. Chỉ cần sử dụng một ứng dụng ví kèm theo là có thể sử dụng được ứng dụng Lighthouse và nhiều ứng dụng khác được viết trên smart contract.
- Không cần phải tải lại trang web mỗi khi chuyển trang (Single Page Application – SPA).
- Áp dụng và kết hợp tốt các công nghệ mới như Smart Contract, React.

- Sử dụng công nghệ IPFS để lưu trữ tập tin. Tận dụng dung lượng còn trống của các thiết bị trong mạng lưới để lưu trữ tập tin phi tập trung, tiết kiệm dung lượng do nếu có hai hay nhiều ảnh giống nhau hệ thống chỉ lưu một ảnh duy nhất và cũng không phải lo lắng sao lưu dữ liệu do các tập tin này đã được sao lưu ra nhiều lần trên toàn mạng lưới. Càng nhiều người dùng hệ thống càng trở nên nhanh hơn.
- Sử dụng điện toán đám mây để triển khai phần front-end của ứng dụng.
- Sử dụng Ethereum Testnet để triển khai phần Smart Contract.

Nhược điểm của ứng dụng:

- Tốc độ tải tập tin khá chậm: do mạng lưới IPFS vừa mới ra đời, còn hạn chế về mặt người dùng nên tốc độ chưa như mong muốn.
- Tốc độ giao dịch chậm: do bản thân Ethereum luôn bị quá tải vì lượng người dùng rất lớn nhưng cơ sở hạ tầng của Ethereum chưa đáp ứng được.
- Chi phí giao dịch cao: do tình trạng quá tải của Ethereum, nên chi phí giao dịch luôn bị đẩy lên mức cao.
- Vẫn phải tin cậy vào bên thứ ba: do chi phí để chạy một nút Ethereum đầy đủ khá cao nên ứng dụng phải nhờ vào một nhà cung cấp thứ ba là infura.io. Và phần front-end vẫn phải nhờ vào Amazon Web Services EC2.
- Chưa sử dụng SSL: ứng dụng vẫn phải sử dụng SSL để mã hóa thông tin, sẽ cập nhật sau này khi triển khai thực tế.
- Chưa có các tính năng tìm kiếm, gửi mail: do Smart Contract chưa hỗ trợ những tính năng này, chi phí rất cao nhưng tốc độ truy suất lại rất thấp. Ứng dụng sẽ bổ sung ngay khi Ethereum hỗ trợ những tính năng trên.
- Chưa hỗ trợ SEO tốt (Tối ưu hóa công cụ tìm kiếm - Search Engine Optimization): do sử dụng React nên các công cụ tìm kiếm như Google không thể đánh chỉ mục được trang.

## II. KẾT QUẢ ĐẠT ĐƯỢC

Ứng dụng website gọi vốn Lighthouse là một trong những ứng dụng đầu tiên áp dụng công nghệ blockchain nhằm giải quyết những vấn đề thực tế. Giúp các giao dịch diễn ra minh bạch, nhanh, an toàn, ẩn danh, an toàn hơn. Từ đó đem lại cho người dùng một trải nghiệm mới tốt hơn. Tuy có những khó khăn nhất định vì đang ở giai đoạn rất sớm của một công nghệ mới, nhưng ứng dụng đã làm tốt vai trò của mình và sẽ là một trong những ứng dụng đi tiên phong và là tiền đề cho các ứng dụng sau này.

Ứng dụng website Lighthouse đã đạt được những mục tiêu đã đề ra ban đầu:

- Tạo các dự án gọi vốn.
- Đăng kí làm thành viên.
- Quản lí thông tin cá nhân.
- Quản lí các dự án đã tạo.
- Ủng hộ các dự án.
- Xem tình trạng của dự án.
- Xem các lịch sử giao dịch trên dự án.
- Rút tiền về ví khi dự án gọi vốn thành công.
- Hoàn trả tiền khi dự án gọi vốn thất bại.
- Lưu trữ tập tin phi tập trung.
- Triển khai dự án lên mạng lưới Ethereum Testnet.

### III. HƯỚNG PHÁT TRIỂN

Trong tương lai tác giả muốn tìm hiểu thêm và phát triển những tính năng sau cho dự án Lighthouse:

- Tìm hiểu sâu hơn về Ethereum cũng như là Bitcoin, sẽ tham gia vào việc đóng góp mã nguồn, phát triển cho hai nền tảng trên.
- Tối ưu hóa tốc độ, cũng như là thêm những tính năng hấp dẫn người dùng như tìm kiếm, gửi mail, ...
- Sử dụng React Server Side Rendering (công cụ tải trang phía máy chủ) để các công cụ tìm kiếm như Google có thể dễ dàng đánh chỉ mục trang, thu hút thêm người dùng.
- Chia sẻ kiến thức cho mọi người, tham gia thảo luận và học hỏi qua các buổi chia sẻ về công nghệ.

## **Tài liệu tham khảo**

- [1] Andreas Antonopoulos (2017). Mastering Bitcoin 2<sup>nd</sup> Edition Programming the Open Blockchain, O'Reilly Media.
- [2] Andreas Antonopoulos, Gavin Wood (2018). Mastering Ethereum 2<sup>nd</sup> Building Smart Contracts and Dapps, O'Reilly Media.
- [4] Jonathan Katz, Yehuda Lindell (2014), Introduction to Modern Cryptography, Second Edition (Chapman & Hall/CRC Cryptography and Network Security Series) 2nd Edition, Chapman and Hall/CRC.
- [5] ‘BLOCKCHAIN’ IS MEANINGLESS. Truy xuất từ: <https://www.theverge.com/2018/3/7/17091766/blockchain-bitcoin-ethereum-cryptocurrency-meaning>.
- [6] Sử dụng style “Tai lieu tham khao”.