**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**Nguyễn Văn Khỏe**

**XÂY DỰNG HỆ THỐNG THƯƠNG MẠI ĐIỆN TỬ DỰA TRÊN NỀN TẢNG BLOCKCHAIN**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Truyền thông và mạng máy tính**

* + 1. **HÀ NỘI - 2018**

**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**Nguyễn Văn Khỏe**

**XÂY DỰNG HỆ THỐNG THƯƠNG MẠI ĐIỆN TỬ DỰA TRÊN NỀN TẢNG BLOCKCHAIN**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Truyền thông và mạng máy tính**

**Cán bộ hướng dẫn: TS. Hoàng Xuân Tùng, TS. Nguyễn Đình Hóa**

**HÀ NỘI - 2018**

**LỜI CAM ĐOAN**

Tôi xin cam đoan những nội dung trong đồ án dưới đây là do chính em thực hiện dưới sự hướng dẫn trực tiếp của TS. Hoàng Xuân Tùng và T.S Nguyễn Đình Hóa. Tất cả tài liệu tham khảo nghiên cứu liên quan đều được ghi nguồn trích dẫn rõ ràng. Nếu có gì sai em xin hoàn toàn chịu trách nhiệm.

Hà Nội, ngày 15 tháng 10 năm 2018

Sinh viên

NGUYỄN VĂN KHỎE

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn sâu sắc của mình tới thầy **Hoàng Xuân Tùng** và thầy **Nguyễn Đình Hóa** thuộc **bộ môn Truyền thông và Mạng máy tính, trường Đại Học Công Nghệ, Đại Học Quốc Gia Hà Nội**, những người đã trực tiếp hướng dẫn em trong suốt quá trình nghiên cứu, hoàn thành đồ án.

Em cũng xin chân thành cảm ơn các Thầy Cô trong khoa Công Nghệ Thông Tin nói riêng và các thầy cô trong trường Đại Học Công Nghệ - Đại Học Quốc Gia Hà Nội nói chung đã dạy dỗ và tận tình chỉ bảo em trong suốt quá trình học tập tại trường.

Cuối cùng em xin gửi lời cảm ơn tới gia đình, bạn bè đã luôn động viên ủng hộ giúp đỡ em để có thể hoàn thành đồ án này.

Một lần nữa, em xin chân thành cảm ơn.

Hà Nội, ngày 15 tháng 10 năm 2018

Sinh viên

NGUYỄN VĂN KHỎE

Mục lục

[Chương 1. Đặt vấn đề 7](#_Toc528196314)

[1.1. Thực tiễn thương mại điện tử, bối cảnh liên quan 7](#_Toc528196315)

[1.2. Tầm quan trọng 7](#_Toc528196316)

[1.3. Mục tiêu nghiên cứu 8](#_Toc528196317)

[Chương 2. Cơ sở lý thuyết (30%) 8](#_Toc528196318)

[2.1. Blockchain 8](#_Toc528196319)

[2.1.1. Định nghĩa 8](#_Toc528196320)

[2.1.2. Đặc tính 9](#_Toc528196321)

[2.1.3. Ứng dụng Blockchain 9](#_Toc528196322)

[2.1.4. Ưu, nhược điểm blockchain với các công nghệ trước đó 9](#_Toc528196323)

[2.2. Hyperledger 9](#_Toc528196324)

[2.2.1. Hyperledger Fabric 9](#_Toc528196325)

[2.2.2. Hyperledger Composer 9](#_Toc528196326)

[2.3. Một số công nghệ ứng dụng 10](#_Toc528196327)

[2.3.1. OAUTH 10](#_Toc528196328)

[2.3.2. Restful API 10](#_Toc528196329)

[2.3.3. AngularJs 10](#_Toc528196330)

[2.3.4. Angular typescript 10](#_Toc528196331)

[2.3.5. Docker 10](#_Toc528196332)

[2.4. Tổng kết chương 10](#_Toc528196333)

[Chương 3. Phân tích thiết kế bài toán xây dựng hệ thống thương mại điện tử dựa trên nền tảng blockchain 10](#_Toc528196334)

[3.1. Tại sao nên xây dựng hệ thống TMĐT ứng dụng công nghệ Blockchain? 10](#_Toc528196335)

[3.1.1. TMĐT truyền thống 10](#_Toc528196336)

[3.1.2. TMĐT ứng dụng blockchain 12](#_Toc528196337)

[3.1.3. So sánh chi tiết 12](#_Toc528196338)

[3.2. Mô tả, xây dựng hệ thống TMĐT ứng dụng Blockchain 14](#_Toc528196339)

[3.2.1. Mô tả chương trình. 14](#_Toc528196340)

[3.2.2. Cấu trúc hệ thống 14](#_Toc528196341)

[3.2.3. Phân tích thiết kế 15](#_Toc528196342)

[Chương 4. Demo (20%) 18](#_Toc528196343)

[4.1. Trình bày thuật toán 18](#_Toc528196344)

[4.2. Quy trình chạy, khởi tạo server 18](#_Toc528196345)

[4.3. Hiển thị kết quả 18](#_Toc528196346)

[Chương 5. Kết luận (10%) 18](#_Toc528196347)

[5.1. Khái quát toàn bộ nội dung đồ án 18](#_Toc528196348)

[5.2. Nhận xét, đánh giá 18](#_Toc528196349)

[5.2.1. Những vấn đề còn tồn tại 18](#_Toc528196350)

[5.2.2. Định hướng phát triển 18](#_Toc528196351)

[Tài liệu tham khảo 18](#_Toc528196352)

**BẢNG CÁC TỪ KHÓA VIẾT TẮT**

|  |  |
| --- | --- |
| **Từ viết tắt** | **Thuật ngữ đầy đủ** |
| TMĐT | Thương mại điện tử |
| HC | Hyperledger Composer |
| HF | Hyperledger Fabric |
| BNA | Mạng lưu trữ thông tin doanh nghiệp |
| Assets | Tài sản, hàng hóa |
| Participants | Các bên liên quan |
| Transactions | Các giao dịchs |
| CSDL | Cơ sở dữ liệu |
| JSON | JavaScript Object Notation |

**DANH MỤC HÌNH ẢNH**

[Figure 1: Cấu trúc business network archive 10](#_Toc528196353)

[Figure 2: Mô hình B2C trong TMĐT truyền thống 11](#_Toc528196354)

[Figure 3: Mô hình TMĐT ứng dụng công nghệ Blockchain 12](#_Toc528196355)

[Figure 4: Cấu trúc hệ thống của ứng dụng 14](#_Toc528196356)

[Figure 5: Sơ đồ mô tả cơ chế xác thực người dùng qua github 16](#_Toc528196357)

[Figure 6: Sơ đồ mô tả cơ sở dữ liệu hệ thống 16](#_Toc528196358)

[Figure 7: Hình vẽ minh họa world-state database 17](#_Toc528196359)

[Figure 8: Hình ảnh minh họa Blockchain database 17](#_Toc528196360)

# Đặt vấn đề

## Thực tiễn thương mại điện tử, bối cảnh liên quan

Thương mại điện tử (E-commerce) là một hình thức kinh doanh thương ại trên cơ sở mạng máy tính toàn cầu. Nó được dự báo là phương thức hoạt động chủ yếu trong nền kinh tế số. Thương mại điện tử trong thời gian gần đây đã được các nước quan tâm và phát triển [3].

Thương mại điện tử chính là nguồn tài nguyên khổng lồ, không những nó không thể bị cạn kiệt như các loại tài nguyên thiên nhiên mà trong kỷ nguyên công nghệ thông tin và viễn thông phát triển như vũ bão, nó ngày càng phong phú và phát triển hơn.

Theo thống kê của Cục viễn thông (Bộ thông tin và truyền thông), tính tới thời điểm tháng 10/2017, cả nước có gần 118 triệu thuê bao di động, trong đó số người sử dụng smartphone chiếm tới 84%. TMĐT tại khu vực ASEAN có tốc độ tăng trưởng internet khoảng 14%/năm. Thống kê của Nielsen cũng cho thấy trung bình mỗi người sử dụng Internet tại Việt Nam chi 160 USD/năm cho TMĐT. Nhờ đó, tốc độc tăng trưởng thương mại điện tử tại Việt Nam tăng khoảng 22%/năm và triển vọng có thể lên tới 30-50%/năm trong giai đoạn 2020-2025 [9].

Với thương mại điện tử mọi thông tin về sản phẩm, khách hàng, doanh nghiệp đều được thường xuyên cập nhập, tạo điều kiện cho khách hàng nắm được những thông tin mới nhất. Việc trao đổi, liên lạc giữa các bên trở lên nhanh chóng, tiết kiệm thời gian, tiền bạc.

## Tầm quan trọng

Trong sự phát triển kinh tế, thương mại điện tử Việt Nam chiếm vai trò vô cùng quan trọng.

TMĐT tạo ra sự thay đổi trong hệ thống sản xuất, như từ mô hình sản xuất hàng loạt sang mô hình sản xuất theo thị trường. Dây chuyền sản xuất còn hợp nhất với các bộ phận tài chính, tiếp thị và bộ phận kinh doanh khác. Ví dụ, trường hợp của công ty Intel, việc sử dụng trang web ERP, các đơn đặt hàng từ phía khách hàng được chuyển tiếp đến người sử dụng hệ thống thiết kế được hỗ trợ bởi máy tính và được chuyển tới dây chuyền sản xuất chỉ trong vài giây [4]. Còn rất nhiều các trường hợp tương tự khác, việc trao đổi thông tin kịp thời trở thành yếu tố quan trọng trong hệ thống liên kết. Giúp những hệ thống này trở lên linh hoạt và dễ thích ứng với sự thay đổi của nhu cầu thị trường. Tối thiểu hóa những dư thừa, tồn đọng trong hệ thống sản xuất.

Thương mại điện tử giúp người tiêu dùng dễ dàng hơn trong việc chọn lựa sản phẩm, dịch vụ. Đây là một kênh phân phối lớn, kênh thông tin bổ sung quan trọng giúp người tiêu dùng dễ tiếp cận sản phẩm, dịch vụ. Chỉ với một chiếc máy tính hay một chiếc smart-phone có kết nối Internet, truy cập vào các sàn giao dịch TMĐT trực tuyến, khách hàng có thể nhận được rất nhiều lời mời chào lựa chọn sản phẩm. Các hê thống TMĐT hiện nay còn tích hợp nhiều công nghệ mới như AI, Machine Learning… sử dụng những thuật toán gợi ý sản phẩm mỗi khi người tiêu dùng sử dụng ứng dụng. Giúp người dùng tiết kiệm thời gian tìm kiếm mặt hàng mình quan tâm. Một điểm ưu không thể không nhắc tới trong TMĐT là sự cản trở về không gian địa lý, thời gian, thiếu thông tin đã được giải quyết một cách khá tối ưu thông qua các dịch vụ TMĐT đi kèm hiện nay có thể kể đến như Grab hay Giaohangtietkiem… Chỉ cần đăng ký tài khoản trên hệ thống và tiến hành đặt hàng, đặt lịch và ngồi nhà chờ đợt, sẽ có nhân viên tới giao mặt hàng tới tận tay bạn. Tiết kiệm rất nhiều thời gian và công sức đi lại của người tiêu dùng.

## Mục tiêu nghiên cứu

Với nguồn nhân lực và tiếp thu nhanh công nghệ mới, Việt Nam được coi là một quốc gia năng động với tỷ lệ người dùng Internet luôn luôn tăng và xếp top đầu các nước sử dụng Internet. Tuy vậy trong TMĐT vấn đề thông tin cá nhân còn được xem nhẹ trong ý thức người Việt. Nhiều công ty doanh nghiệp đang thu thập số lượng lớn dữ liệu người dùng cá nhân. Các dữ liệu này thường là các dữ liệu rất nhạy cảm, liên quan tới quyền riêng tư của người tiêu dùng. Các mối lo ngại về bảo mật thông tin cho người dùng trở thành vấn đề lớn cho các doanh nghiệp đó, không ai có thể đảm bảo hệ thống của mình bảo mật hoàn hảo.

Ngoài ra, tại thị trường TMĐT ở Việt Nam, quyền lợi của người tiêu dùng, các đại lý phân phối vẫn xếp bên dưới quyền lợi của doanh nghiệp. Ban đầu, doanh nghiệp tiến hành xây dựng sàn giao dịch của họ, thiết lập các quy định, điều lệ phù hợp với họ nhất. Cũng không thể phủ nhận họ rất coi trọng ý kiến, quyền lợi bên tiêu dùng và bên đại lý phân phối vì đây chính là nguồn lợi nhuận của họ. Tuy nhiên xét về phương diện đại lý phân phối, họ không trực tiếp được phân phối sản phẩm mà phải trung chuyển qua bên thứ ba là doanh nghiệp, kèm theo đó chịu những quy định điều lệ mà bên doanh nghiệp đề ra. Phương diện người tiêu dùng thì không được mua với đúng mức giá chuẩn mà bên đại lý phân phối ban đầu đưa ra…

Trong khuôn khổ đồ án này, em xin đề xuất xây dựng một hệ thống TMĐT mới do em phát triển dựa trên việc sử dụng các ưu điểm của công nghệ Blockchain. Cho phép áp dụng những thuật toán mới giúp cải thiện tính bảo mật thông tin, quyền lợi giữa các bên tiêu dùng, phân phối sản phẩm được đảm bảo một cách tối ưu hơn so với TMĐT hiện nay.

# Cơ sở lý thuyết (30%)

### TMĐT

TMĐT hiện nay được tổ chức theo mô hình business-to-consumer (B2C). Mô hình này mô tả các hoạt động kinh doanh phục vụ người tiêu dùng cuối (end-users) với các sản phẩm, dịch vụ. Một cá nhân truy cập trang web của tổ chức trước khi họ mua sản phẩm. Thông qua các thao tác click chuột, nhấp vào các liên kết mua sắm thì hệ thống trang web sẽ gửi cho họ hàng trăm sản phẩm, dịch vụ cung cấp bởi các thương hiệu hàng đầu. Giao dịch mua được thực hiện như thể họ giao dịch trực tiếp với các người bán, các thương gia. Các giao dịch mua được gửi tới người mua và tổ chức giới thiệu mặt hàng sẽ nhận được hoa hồng trên mỗi giao dịch thành công [6] (>> trich dan ro ten tac gia, ten sach<< va move len ben tren) .

Để có thể đưa các sản phẩm, dịch vụ của mình đến người tiêu dùng thông qua website thì các thương gia cần phải chi trả một khoản chi phí cho tổ chức quản lý website.

Ta có thể khái quát mô hình B2C như sau:

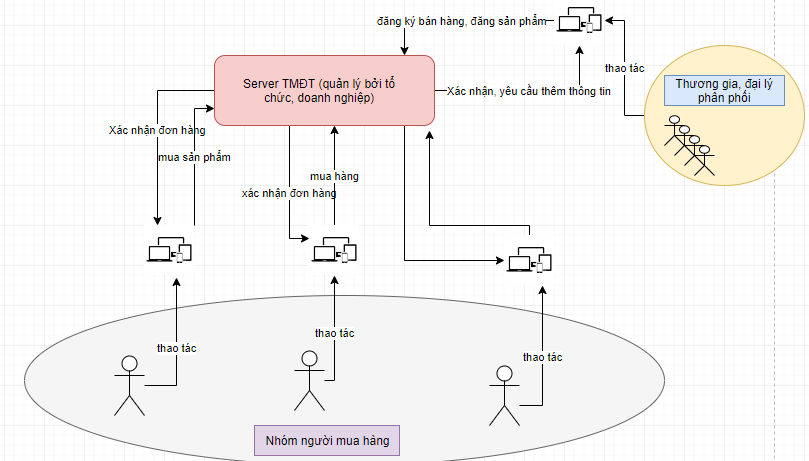


Figure 1: Mô hình B2C trong TMĐT truyền thống

Ta thấy rõ được ở TMĐT truyền thống chia làm ba nhóm đối tượng chính: Người tiêu dùng (bên mua), tổ chức kinh doanh quản lý website, thương gia (người bán).

Bên nhóm người tiêu dùng, họ tiến hành thao tác với ứng dụng trên website tiến hành chọn lựa đặt hàng những mặt hàng mình ưa thích. Những đơn hàng, thông tin thay đổi liên quan tới người dùng sẽ được đóng gói lại và tạo lên một request chứa gói thông tin này cho bên server, nhóm tổ chức quản lý server này họ sẽ tiến hành lưu trữ, xác nhận các đơn hàng và thay đổi tùy theo những yêu cầu hợp lệ từ bên khách hàng.

Bên nhóm thương gia, đại lý phân phối họ tiến hành đăng ký thương hiệu trên website, liệt kê những mặt hàng mình muốn bán, đăng sản phẩm của mình lên website. Tương tự như bên nhóm người tiêu dùng, tất cả các thông tin này sẽ được đóng gói lại được gửi qua một request cho bên server. Bên server sẽ tiến hành tạo quy định hợp đồng, tiến hành lưu trữ đăng sản phẩm của bên đại lý muốn bán…

Về quyền thì ở đây tổ chức quản lý hệ thống website có quyền cao nhất, có thể quyết định tất cả các giao dịch có được phép thành công hay không. Họ có thể truy cập vào thông tin tài khoản cả bên mua và bên bán, quy định tất cả các điều lệ mà bên bán và bên mua phải tuân theo.

## Blockchain

### Định nghĩa

Một mạng blockchain có thể định nghĩa như một sổ kế toán bất biến (immutable ledger) để ghi lại tất cả các giao dịch, được duy trì trong hệ thống phân phối giữa các bên không tin cậy. Mỗi bên lưu trữ một bản sao sổ kế toán. Các bên thực hiện giao thức đồng thuận để xác thực giao dịch, nhóm chúng thành một khối và xây dựng các chuỗi băm trên mỗi khối [1].

### Đặc tính

#### Smart contract

#### Consensus

#### Distribution

#### Toàn vẹn dữ liệu

#### Bảo mật, tin cậy

#### Minh bạch, bất biến

### Ứng dụng Blockchain

### Ưu, nhược điểm blockchain với các công nghệ trước đó

## Hyperledger

Dự án Hyperledger ([www.hyperledger.org](http://www.hyperledger.org)) là kết quả của sự nỗ lực cộng tác để tạo ra một khung phân phối sổ kế toán mã nguồn mở, thúc đẩy công nghệ blockchain. Nó nhằm mục đích nâng cao công nghệ blockchain bằng cách xác định và thực hiện một nền tảng tiêu chuẩn mở ngành công nghiệp để phân phối sổ cái (distributed ledgers), có thể biến đổi cách thức giao dịch kinh doanh được tiến hành trên toàn cầu. Dự án được tổ chức bởi Linux Foundation, bao gồm các nhà lãnh đạo về tài chính, ngân hàng, Internet of Thing (IoT), chuỗi cung ứng, sản xuất và công nghệ [5].

Có thể coi Hyperledger là một hệ sinh thái được phát triển để minh bạch, công khai khai sáng tiềm năng của công nghệ Blockchain ứng dụng vào các lĩnh vực trong doanh nghiệp.

Hyperledger không hỗ trợ Bitcoin hoặc bất kỳ cryptocurrency khác.

Hyperledger phát triển một chiến lược gọi là Hyperledger Umbrella với nhiều dự án khác nhau.

### Hyperledger Fabric

Hyperledger Fabric là một mô đun và hệ thống phân tán có thể mở rộng để chạy mạng blockchain cho phép[2]. Hỗ trợ các giao thức đồng thuận blockchain có thể cắm vào để đảm bảo rằng các giao dịch được xác thực theo chính sách bởi những người tham gia mạng lưới kinh doanh được chỉ định.

Hyperledger Fabric cũng là hệ thống blockchain đầu tiên chạy trên tiêu chuẩn ứng dụng phân tán, ngôn ngữ lập trình phục vụ mục đích chung, không phụ thuộc vào một ngôn ngữ tiền điện tử cụ thể nào.

#### Cấu trúc cơ sở dữ liệu trên framework Hyperledger Fabric (ví blockchain):

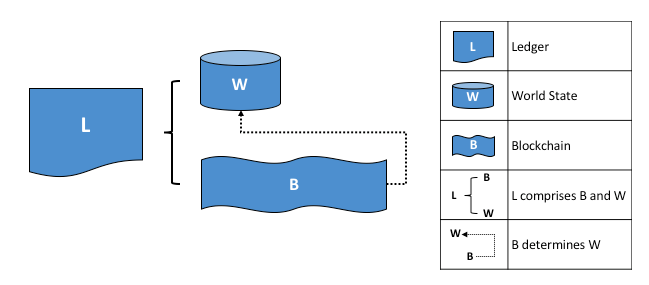


Figure 2: Sơ đồ mô tả cơ sở dữ liệu hệ thống

Cơ sở dữ liệu tích hợp blockchain này bao gồm hai phần chính: **world-state** database and **blockchain** database.

Đầu tiên, world-state database chứa giá trị hiện tại của một tập hợp trạng thái của sổ kế toán. Mặc định, trạng thái của sổ cái được biểu thị dưới cặp key-value. Cũng như cái tên world state – trạng thái trái đất là luôn luôn vận động, các cặp key-value này có thể được tạo mới, cập nhật và xóa. Lấy một ví dụ cụ thể trong ứng dụng em phát triển thì khi người dùng đăng một món hàng lên sàn giao dịch. Thông tin chi tiết về món hàng, định danh của nó sẽ được lưu trữ vào world-state database. Khi người dùng này muốn sửa đổi thông tin về món hàng này hoặc gỡ bỏ món hàng vì một lý do nào đó không muốn bán nữa thì thao tác này có thể thực hiện, dữ liệu trong database sẽ được tùy biến thay đổi.

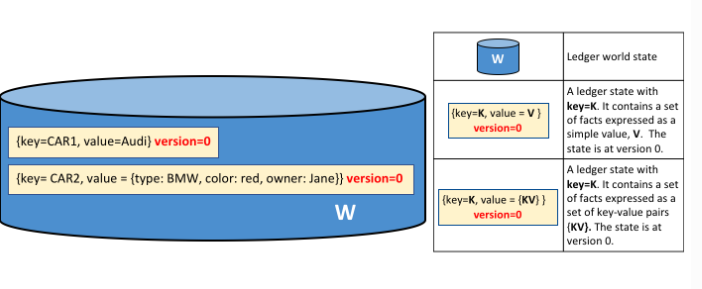


Figure 3: Hình vẽ minh họa world-state database

Thứ hai, blockchain database – nhật ký giao dịch ghi lại tất cả các trạng thái thay đổi của world-state database. Các giao dịch được thu thập sẽ được lưu vào block nối tiếp vào chuỗi blockchain trong mạng. Cấu trúc này khác hoàn toàn với cấu trúc bên trong world-state database vì một khi dữ liệu được thêm vào thì nó gần như không thể thay đổi được. Nó là một chuỗi bất biến của các khối, các khối chứa một tập hợp các giao dịch có thứ tự.

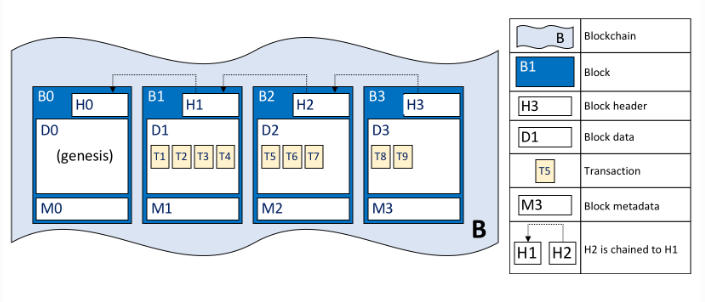


Figure 4: Hình ảnh minh họa Blockchain database

Sự kết hợp hai database này lại thu được một sổ cái hay còn gọi với thuật ngữ Ledger.

### Hyperledger Composer

Hyperledger Fabric được sử dụng để chạy mạng lưới kinh doanh trong khi đó Hyperledger Composer được sử dụng để triển khai các hợp đồng thông minh hoặc các chức năng xử lý giao dịch chạy trên mạng blockchain [10].

Mục tiêu chính mà Hyperledger Composer được tạo ra là tăng tốc thời gian xử lý. Nhanh chóng mô hình hóa mạng lưới kinh doanh hiện tại, chứa các tài sản, các giao dịch liên quan tới chúng, tài sản là hàng hóa, dịch vụ hoặc tài sản vô hình hoặc hữu hình. Là một phần trong mạng doanh nghiệp.

#### Cấu trúc

##### Business network archive

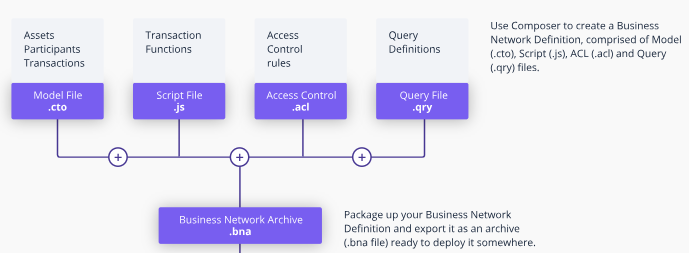


Figure 5: Cấu trúc business network archive

##### Hyperledger Composer Playground

##### REST API

## Một số công nghệ ứng dụng

### OAUTH

### Restful API

### AngularJs

### Angular typescript

### Docker

## Tổng kết chương

# Phân tích thiết kế bài toán xây dựng hệ thống thương mại điện tử dựa trên nền tảng blockchain

## Ứng dụng Blockchain vào TMĐT

### TMĐT tích hợp blockchain

Mô hình TMĐT này em thiết kế theo mô hình peer-to-peer (P2P). P2P nhấn mạnh vào các hoạt động mua bán được thực hiện trực tiếp giữa người mua, người bán thông qua website. Không như hệ thống TMĐT truyền thống thì bên tổ chức quản lý server chiếm mọi quyền trong tất cả các giao dịch. Ở đây vai trò của người tiêu dùng (bên mua) và người bán hay các thương gia có quyền lớn nhất (bên bán). Bên quản lý server không có hoặc có rất ít quyền trong việc thiết lập các giao dịch. Quyền lợi giữa bên bán và bên mua được đẩy lên cao nhất.

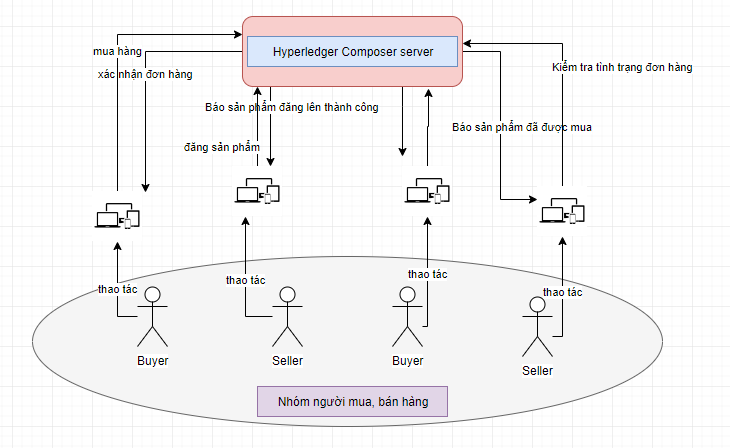


Figure 6: Mô hình TMĐT ứng dụng công nghệ Blockchain

Ở đây nhóm bán và nhóm tiêu dùng được gộp chung lại một nhóm. Họ có quyền cao nhất trong các giao dịch. Gần như peer-to-peer, giao dịch trực tiếp với nhau. Khác với server truyền thống thì server trong ứng dụng TMĐT áp dụng blockchain này sử dụng Hyperledger composer server. Như được trình bày ở phần <<TODO>> thì server này lưu trữ thông tin các giao dịch trên mạng blockchain và state DB <<TODO>>.

Để đi sâu vào khác biệt giữa hai sàn giao dịch này và lợi ích mà giao dịch TMĐT ứng dụng công nghệ Blockchain đem lại, chúng ta đến với phần so sánh chi tiết sau đây.

### Lợi ích khi ứng dụng

#### Tổ chức quản lý hệ thống

*Điểm chung*: Doanh nghiệp, tổ chức hay một cá nhân đứng ra quản lý hệ thống. Hệ thống website lưu trữ tất cả các giao dịch, thông tin khách hàng, đối tác bán hàng…

*Khác biệt*: So với TMĐT truyền thống thì TMĐT ứng dụng blockchain có điểm khác biệt cơ bản ở vai trò của con người đứng ra tổ chức hệ thống được giảm bớt đi.

Ví dụ để quản lý một hệ thống TMĐT thì doanh nghiệp, tổ chức phân ra rất nhiều các ban, các phòng… Mỗi ban, mỗi phòng chiếm một vai trò nhất định như: duy trì, quản lý máy chủ, tìm kiếm đối tác, mặt hàng, quản lý thu chi, hành pháp…

Để quản lý một hệ thống TMĐT ứng dụng Blockchain thì rất nhiều ban, phòng… có thể được bỏ qua. Có thể chỉ cần ít nhất bộ phận duy trì, quản lý máy chủ là đủ để duy trì cả hệ thống. Nhu cầu nhân lực được giảm đi rất nhiều.

#### Bảo mật

Sàn giao dịch TMĐT truyền thống dễ bị khai thác dữ liệu. Các lỗ hổng chính là mục tiêu tấn công của các hacker. Họ có thể lợi dụng chúng để tiến hành làm giả, thay đổi thông tin các giao dịch. Tiến hành mua nhiều sản phẩm mà không mất bất kỳ chi phí nào. Hay ăn cắp thông tin của người dùng khác…

TMĐT ứng dụng công nghệ Blockchain là một sàn giao dịch ứng dụng công nghệ mới, tính bảo mật cao hơn. Thông tin các giao dịch, định danh người dùng một khi được lưu vào mạng blockchain thì cực kỳ khó để có thể thay đổi. Từ đó duy trì tính xác thực của thông tin.

#### Hợp đồng điện tử

Trong sàn giao dịch TMĐT truyền thống: Khi bên doanh nghiệp, tổ chức quản lý hệ thống tiến hành ký kết hợp đồng kinh doanh với các thương gia, các đại lý cung cấp mặt hàng. Thì để tạo ra hợp đồng điện tử này thì phải mất chi phí cho bên pháp lý biên soạn và có bên thứ ba đứng ra thực thi. Ở đây phải mất thời gian và tốn kém, nhiều khi còn thiếu tính minh bạch trong các khâu để hoàn thành hợp đồng. Những hợp đồng xảy ra sự cố thì lại phải dựa vào hệ thống pháp lý để giải quyết.

Với sàn giao dịch TMĐT ứng dụng công nghệ Blockchain thì hợp đồng giữa các bên được gọi với thuật ngữ smart-contract (hợp đồng thông minh), ở đó các bên tham gia phải chấp thuận các điều khoản trong hệ thống được lập trình viên thiết lập từ trước. Toàn bộ đoạn mã được thực thi trên hệ thống sổ cái Blockchain. Trong đó, quy định tất cả các điều khoản và hình phạt tương đương. Sau khi hệ thống khởi chạy thì ký kết việc hợp đồng giữa các bên không cần sự can thiệp của bên tạo hợp đồng nữa. Quyền lợi giữa bên mua và bên bán được đảm bảo, độ tin cậy cao, ít xuất hiện lỗi ngoài ý muốn. Mục đích chính ở đây là cho phép bên mua và bên bán có thể giao dịch trực tiếp với nhau thông qua internet mà không phải lệ thuộc quá nhiều vào bên trung gian.

#### Đồng thuận

Với TMĐT truyền thống: thì có một hệ thống phân cấp rõ ràng. Doanh nghiệp, tổ chức quản lý sàn giao dịch (gọi tắt bên A) là bên có quyền cao nhất. Các giao dịch, thông tin người dùng hay người bán (các thương gia, các đại lý) (gọi tắt bên B) có thể bị bên A thay đổi. Bên A có thể hủy hay áp đặt các điều lệ mới để đặt lợi nhuận kinh doanh của mình. Để đăng một sản phẩm lên sàn thì người bán phải chịu một khoản phí hoa hồng cho bên A. Mức giá mà người mua mua trên sàn giao dịch này lớn hơn mức giá chuẩn mà người bán đặt ra. Từ đây bên A có thể thu lợi nhuận từ cả bên bán và bên mua. Từ đó thấy được bên chịu thiệt luôn là bên B.

Ngược lại với TMĐT ứng dụng Blockchain thì đề cao tính ngang hàng (peer-to-peer). Trong mọi giao dịch bên A không có vai trò gì. Giao dịch chỉ được phép thực hiện khi bên mua và bên bán chấp thuận, đảm bảo thuận mua vừa bán. Mức giá sản phẩm của bên mua phải chi trả luôn bằng đúng mức giá mà bên bán đề ra. Ở đây quyền lợi người tiêu dùng và người bán hàng được đảm bảo một cách tối ưu.

*Tóm lại*, khi ứng dụng công nghệ Blockchain vào TMĐT lợi ích đem lại vô cùng to lớn. Giảm bớt chi phí, nhân lực, thời gian, tiền bạc… Thông tin được bảo mật tốt hơn.

## Mô hình hóa hệ thống TMĐT ứng dụng Blockchain

Ứng dụng mang đầy đủ các chức năng cơ bản mà một sàn giao dịch thương mại điện tử hiện nay đang có. Có thể kể đến như: đăng nhập vào ứng dụng, mua sản phẩm, bán sản phẩm, giỏ hàng, thanh toán, nạp tiền, lịch sử giao dịch.

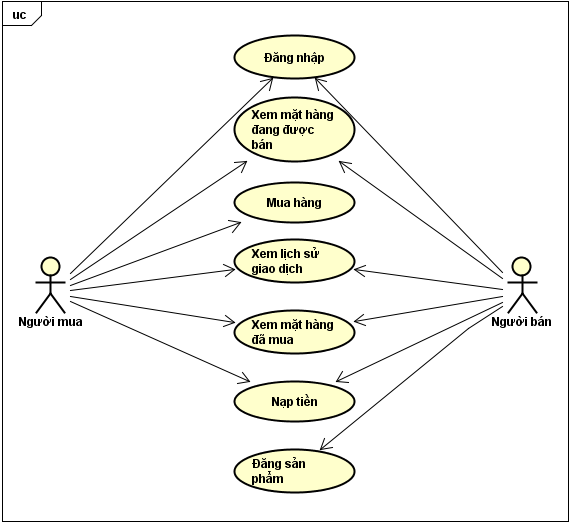


Figure 7: Mô hình chức năng hệ thống

### Phân tích hệ thống

Ứng dụng được phát triển theo mô hình client-server. Giao diện client trên nền tảng website. Ứng dụng được đặt tên là Exchange App.

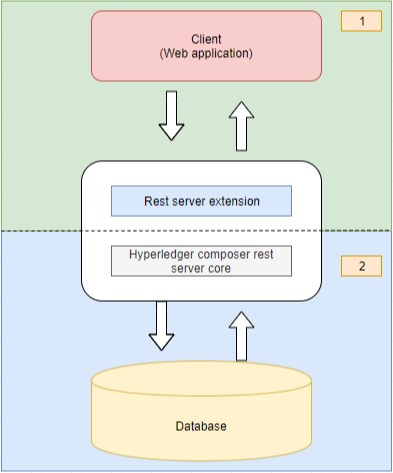


Figure 8: Cấu trúc hệ thống của ứng dụng

Hệ thống được chia ra hai phần chính:

* Phần phát triển: client và rest server extension (phần 1 trong hình).
* Phần công cụ sử dụng: Hyperledger composer rest server core và cơ sở dữ liệu tích hợp blockchain (phần 2 trong hình).

Về công cụ sử dụng: Hyperledger composer server core được miêu tả chi tiết ở phần <<TODO bên trên>>. Cơ sở dữ liệu tích hợp công nghệ blockchain được miêu tả rõ ở phần ledger fabric <<TODO>>

Về phần phát triển, bên client được phát triển trên nền tảng web application cung cấp giao diện web cho người dung tương tác với ứng dụng. Rest server extension là rest service được phát triển khả rộng trên công cụ Hyperledger Composer server.

### Thiết kế

#### Hyperledger composer rest server extension

##### Model

* Enumerated type

Tập các giá trị, thuộc tính cụ thể đã được biết, mô hình kiểu liệt kê.

Trong Exchange App khai báo một số enum như sau:

* ProductType: các loại sản phẩm

|  |  |
| --- | --- |
| * ProductType | Mô tả |
| SMART\_PHONE | Điện thoại thông minh |
| VEGETABLE | Rau, củ, quả |
| CAR | Xe ô tô |
| CLOTHES | Quần áo |

* ProductStatus: tình trạng sản phẩm

|  |  |
| --- | --- |
| ProductStatus | Mô tả |
| SELLING | Đang được bày bán |
| BOUGHT | Đã được mua |

* Concept

Thực thể tồn tại trong mô hình kinh doanh, nhưng không phải tài sản, các bên tham gia.

Trong ứng dụng, em sử dụng một concept là Address.

|  |  |  |
| --- | --- | --- |
| Address | Kiểu dữ liệu | Mô tả |
| country | String | Tên quốc gia |
| city | String | Tên thành phố, tỉnh |
| street | String | Tên phường, huyện |

* Assets

Assets ở đây có thể là bất cứ tài sản nào có giá trị như: một chiếc điện thoại nokia, một chiếc áo, một kilogam xoài... Tất cả chúng là những thứ có thể được đem lên trao đổi.

Trong Exchange App, asset ở đây là những món hàng mà người dùng muốn trao đổi mua hoặc bán được trừu tượng hóa với tên Product.

|  |  |  |
| --- | --- | --- |
| Thuộc tính của Product | Kiểu dữ liệu | Mô tả |
| productName | String | Tên mặt hàng |
| productType | String | Loại sản phẩm |
| pricePerUnit | Long | Giá trên từng đơn vị sản phẩm (đơn vị VNĐ) |
| amount | Long | Số lượng |
| imageBase64Encode | String | Mã hóa base64 của hình ảnh mặt hàng |
| description | String | Mô tả mặt hàng |
| cartStatus | String | Tình trạng mặt hàng trong giỏ hàng |
| productStatus | ProductStatus | Tình trạng mua, bán mặt hàng |
| owner | Collector | Chủ mặt hàng |
| ownerImage | String | Ảnh chủ mặt hàng |

* Participants

Một participant là một bên tham gia mô hình mạng.

Tất cả các bên được xác thực bởi oauth <<TODO: 3.2.2.1.1>> trước khi họ có quyền truy cập vào ứng dụng. Khi người dùng này được xác thực, họ được chỉ định một thẻ ID blockchain bởi Membership Services [7]. Ở trong, hệ thống này bên tham gia là người mua và người bán. Người dùng vừa có thể là người mua vừa có thể là người bán. Một cá nhân có thể đăng ký một, hai hoặc ba tài khoản. Mỗi tài khoản được gán cho một ID duy nhất. Mỗi người tham gia là chủ sở hữu dữ liệu và yêu cầu dữ liệu. Người yêu cầu dữ liệu là người gửi được xác thực yêu cầu truy cập dữ liệu cụ thể của chủ sở hữu.

Với ứng dụng Exchange App, nó bao gồm người mua hàng, người bán hàng, được trừu tượng hóa với tên là Collector.

|  |  |  |
| --- | --- | --- |
| Thuộc tính của Collector | Kiểu dữ liệu | Mô tả |
| email | String | Email người dùng login qua github |
| accountBalance | Long | Số dư tài khoản |
| firstName | String | Tên người dùng |
| lastName | String | Họ người dùng |
| phoneNumber | String | Số điện thoại người dùng |
| address | Address | Địa chỉ người dùng |
| coverPhoto | String | Mã hóa base64 của ảnh đại diện người dùng |

* Transactions

Khi một asset được động tới như mua hay thay đổi trạng thái, thì tất cả những thay đổi đó sẽ làm thay đổi trạng thái của sổ cái blockchain. Sự tương tác trong model của Hyperledger Composer đều được gọi là transactions, dịch nôm na trong tiếng việt là các phiên giao dịch.

Trong Exchange App thì có một số transaction như sau: chuyển nhượng mặt hàng khi giao dịch mua bán thành công (Trade), nạp tiền (IncreaseBalance), chuyển sản phẩm sang giỏ hàng (MoveProToCart).

|  |  |  |
| --- | --- | --- |
| Thuộc tính của Trade | Kiểu dữ liệu | Mô tả |
| oldOwner | Collector | Chủ sở hữu cũ của sản phẩm |
| oldOwnerName | String | Tên chủ sở hữu cũ của sản phẩm |
| product | Product | Sản phẩm được trao đổi |
| productName | String | Tên sản phẩm được trao đổi |
| newOwner | Collector | Chủ sở hữu mới của sản phẩm |
| newOwnerName | String | Tên chủ sở hữu mới của sản phẩm |
| image | String | Mã hóa base64 của ảnh sản phẩm |

##### Script file

Triển khai chi tiết các nghiệp vụ logic của các transaction để tạo nên một smart-contract như được mô tả ở <<TODO>>. Nêu rõ các điều kiện, điều khoản cụ thể, các giá trị được thiết lập. Ví dụ: chuyển đổi thông tin mặt hàng giữa hai khách hàng A và B, A phải là chủ sở hữu của mặt hàng Z, mặt hàng Z còn trong tình trạng còn hàng, B thì phải đủ số tiền để thanh toán mặt hàng…

Trong Exchange App, mô tả chi tiết các logic nghiệp vụ của các transaction.

Transaction Trade:

* Đổi quyền sở hữu sản phẩm cho người mua.
* Chuyển tình trạng sản phẩm từ đang bán sang đã được mua.
* Chuyển trạng thái giỏ hàng của sản phẩm sang tình trạng phi giỏ hàng.
* Tính tổng số tiền của sản phẩm (số lượng x giá thành trên từng đơn vị sản phẩm), và trừ số tiền đó vào tài khoản của người mua đồng thời cộng vào số dư tài khoản của người bán.
* Cập nhật trạng thái sổ kế toán blockchain. Bước này được coi là quan trọng nhất.

Transaction MoveProToCart:

* Đổi trạng thái giỏ hàng mặc định của sản phẩm khi được bày bán bằng email người mua tạm thời (chưa mua, mới chỉ tích sản phẩm vào giỏ hàng).
* Cập nhật trạng thái sổ kế toán blockchain.

Transaction IncreaseBalance:

* Cộng vào số dư tài khoản của người dùng với số tiền được nạp.
* Cập nhật trạng thái sổ kế toán blockchain.

##### Access control

Trong một mạng lưới kinh doanh không phải ai cũng có quyền truy cập mọi thứ. Ví dụ, người mua chỉ xem được thông tin của mình và những món hàng mình đã mua, không xem được thông tin của tất cả người dùng, hay được phép sửa thông tin món hàng người khác đăng bán.

Trong Exchange App, em có khai báo một số quyền truy cập dữ liệu đối với mỗi người dùng khác nhau.

* Wholesaler: Cho phép mọi người dùng được phép xem những sản phẩm đang được bày bán, trừ những sản phẩm mình đăng lên sàn giao dịch. Người dùng chỉ có quyền đọc dữ liệu (READ), với điều kiện là chủ sở hữu sản phẩm khác với người dùng đang sử dụng. <<TODO: trong code còn cho UPDATE>>>
* CollectorAssets: Chặn không cho phép người dùng khác có thể sửa thông những sản phẩm mà họ không sở hữu. Chặn tất cả các quyền CREAT, READ, UPDATE, DELETE.
* ProductViews: Cho phép người dùng được tạo, xem và sửa thông tin sản phẩm mà họ sở hữu. Người dùng có tất cả các quyền truy cập dữ liệu: CREAT, READ, UPDATE, DELETE.
* MakeTrade\_1: Cho phép người dùng tạo, xem phiên giao dịch mua sản phẩm. <<TODO: trong code nên để là CREATE, READ>>
* makeListProToCart: Cho phép người dùng xem và thêm sản phẩm đang được bày bán vào giỏ hàng. Người dùng truy cập dữ liệu với quyền READ, UPDATE.
* CollectorsSeeThemeselves: Cho phép người dùng tạo, xem, cập nhật hay xóa thông tin của mình trên ứng dụng. Truy cập dữ liệu với quyền CREATE, READ, UPDATE, DELETE.
* CollectorsSeeOwnTransactions: Mọi người dùng đều có quyền xem tất cả thông tin các phiên giao dịch.
* DenyHistorian: <<TODO>>
* NetworkUpdate: Cho phép hệ thống cập nhật tất cả các quyền truy cập.
* SystemACL: Cho phép hệ thống cập nhật tất cả các quyền truy cập.
* NetworkAdminUser: Người dùng admin có quyền truy cập tất cả dữ liệu người dùng.
* NetworkAdminSystem: Người dùng admin có quyền truy cập tất cả dữ liệu hệ thống.

##### Query file

File chứa các câu lệnh truy vấn dữ liệu sql, có thể kèm các điều kiện hay hoàn cảnh cụ thể.

Với Exchange App như có một số lệnh sql query phục vụ việc tìm kiếm những sản phẩm đang trong tình trạng còn bán, đã được mua, những sản phẩm mà người dùng đã mua hoặc đang bán, danh sáchcác mặt hàng trong giỏ hàng...

* availableProducts: Lấy tất cả các sản phẩm đang được bày bán trừ sản phẩm mà người dùng đang bán.
* myProductSelling: Lấy tất cả các sản phẩm mà người dùng đang bày bán.
* myProductBought: Lấy tất cả các sản phẩm mà người dùng đã mua.
* listProductsInMyCart: Lấy tất các các sản phẩm trong giỏ hàng người mua.

|  |  |  |  |
| --- | --- | --- | --- |
| Participant | Participant accessed | Assets accessed | Transactions accessed |
| Người mua | Người mua ® | Product (RU) | Wholesaler, CollectorAssets,  ProductViews,  MakeTrade\_1,  makeListProToCart,  CollectorsSeeThemeselves,  CollectorsSeeOwnTransactions |
| Người bán | Người bán ® | Product © |  |
| Admin | ALL (CRUD) | ALL (CRUD) | ALL (CRUD) |

##### Business network archive (BNA)

Business network archive là một file nén lưu trữ thông tin về các tài sản, hàng hóa, và các phiên giao dịch liên quan tới chúng. Được đóng gói lại và xuất ra dưới dạng file nén để triển khai trên ứng dụng web.

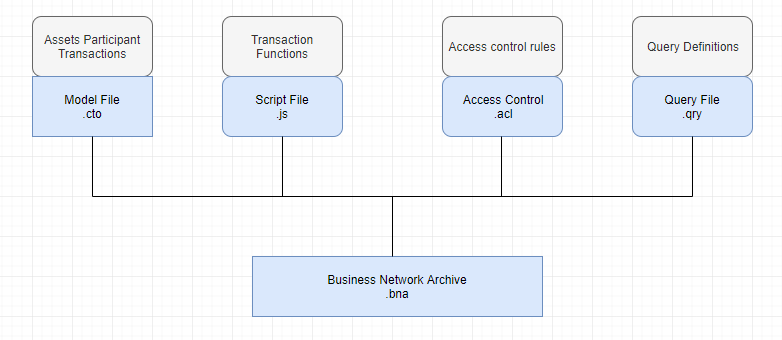
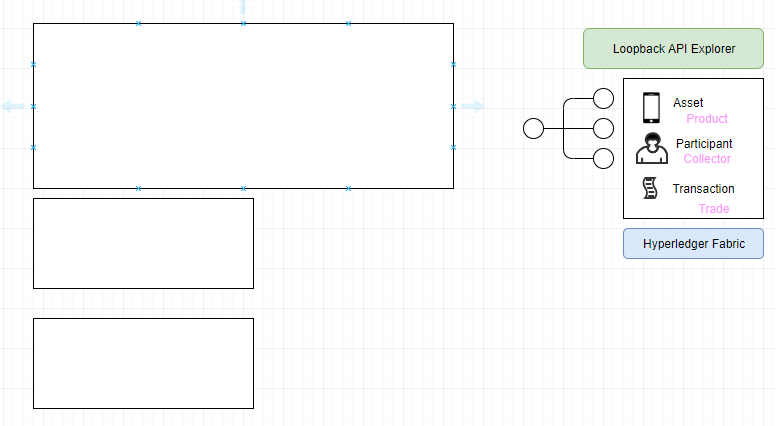


Figure 9: Mô hình Business network archive

Hyperledger Composer đóng gói dữ liệu từ bốn file trên và một số file cấu hình cần thiết thành một file (định dạng .bna). Sử dụng file này kết hợp với ID Cards (bao gồm thông tin người đăng nhập và xác thực) ta sẽ kết nối tới một sổ kế toán phân phối (distributed ledger) trên nền Fabric. Nơi đó dữ liệu được lưu trên nền tảng Blockchain.

##### Rest API support

Khi hệ thống được chạy, để truy cập và cập nhật dữ liệu bên trong sổ kế toán, HC hỗ trợ phương thức REST. Ví dụ: getAvailbleProducts (thông tin tất cả các món hàng đang được rao bán) hay getAllTransactions (thông tin giao dịch toàn hệ thống) … Dữ liệu trả về dưới dạng JSON object.



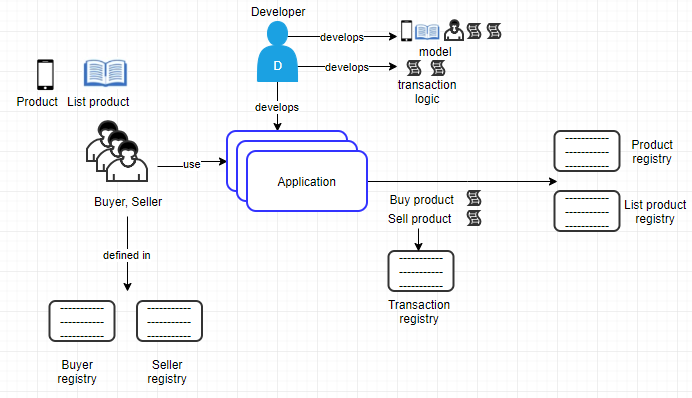


Figure 10: Mô hình tổng quan xây dựng ứng dụng

#### Client

Phát triển trên nền web với framework Angular typescript. Sử dụng bootstrap 4 để thiết lập màu sắc, căn chỉnh giao diện và các annotations thông qua javascript.

Xây dựng một giao diện cho sàn TMĐT với các chức năng được liệt kê ở hình <<TODO>>

Routing giữa các phần (trang chủ, giỏ hàng, nạp tiền…)

Client giao tiếp với server qua giao thức API HTTP. Giúp việc kết nối dữ liệu qua REST API.

#### Liên kết client và server

<<image>>

#### Cơ chế đa người dùng

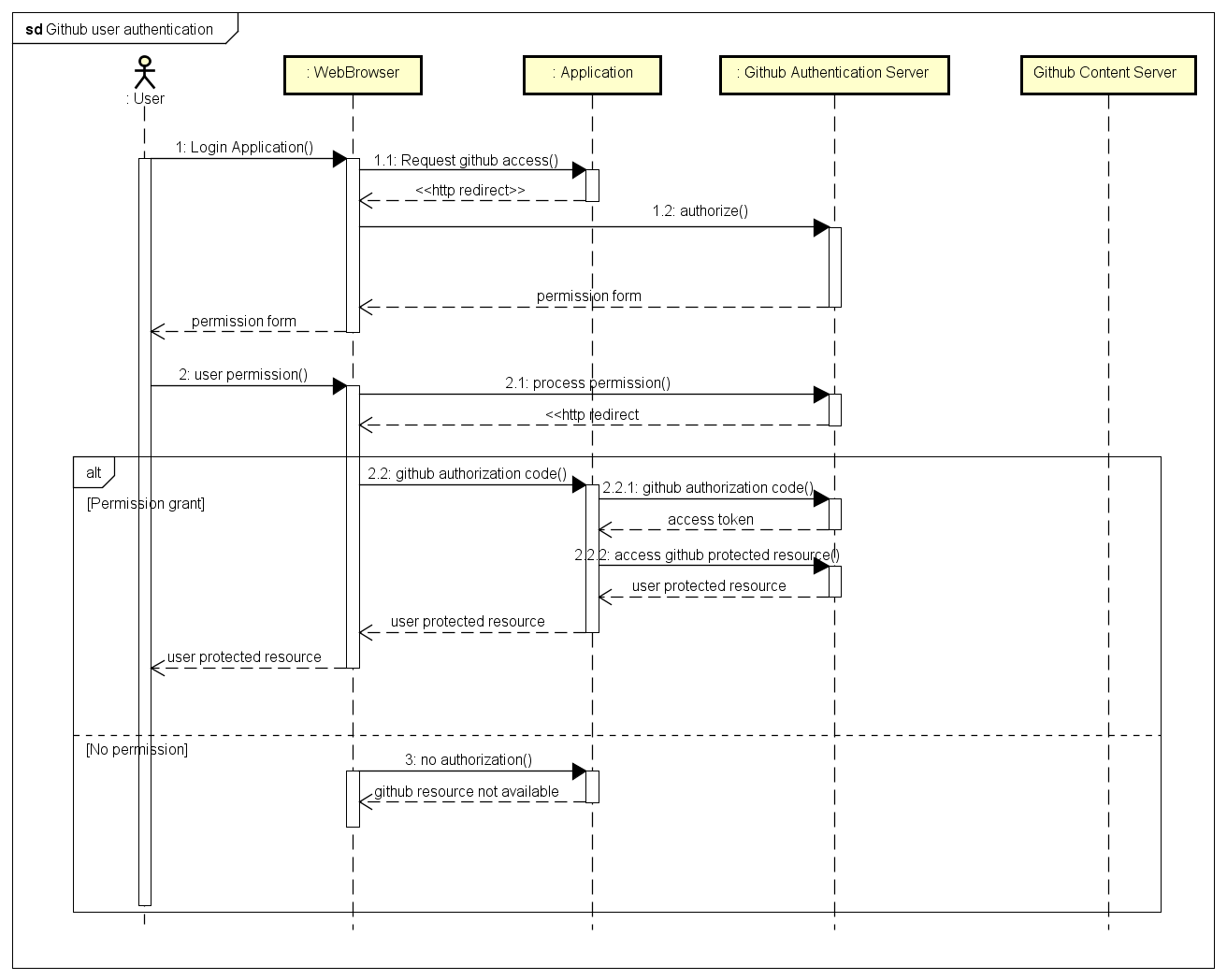
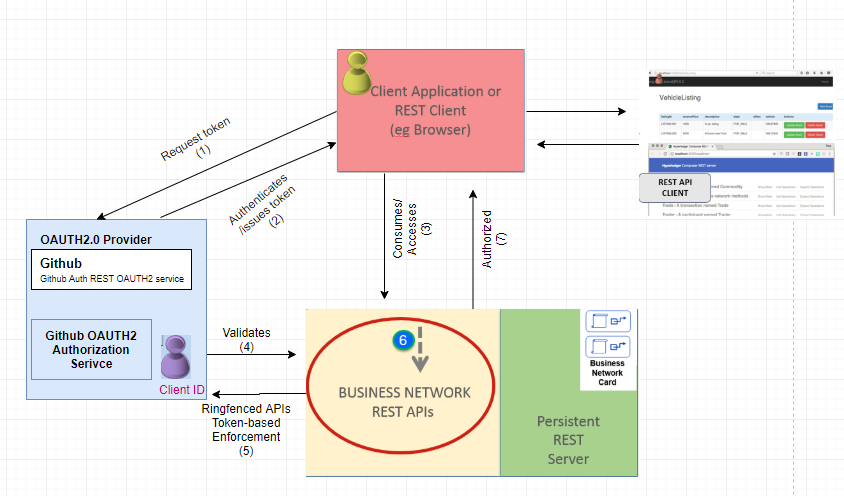


Figure 11: Sơ đồ mô tả cơ chế xác thực người dùng qua github



# Demo (20%)

## Chuẩn bị

(Chú ý: Các bước cài đặt được viết bên dưới có tham khảo từ trang <https://hyperledger.github.io>)

### Cài đặt

#### Công cụ tiên quyết

* Hệ điều hành: Ubuntu Linux 18.04.1 LTS.
* Docker Engine: phiên bản …
* Docker-Compose: phiên bản …
* Nodejs phiên bản …
* Npm …
* Git: …
* Python: …
* IDE: Visual Studio Code

#### **Cài đặt Hyperledger Composer**

Cài đặt công cụ CLI thao tác trên terminal của ubuntu:

* composer-cli: công cụ cần thiết nhất bao gồm tất cả các thao tác cần thiết trên terminal. Cài qua dòng lệnh

*sudo npm install -g* [*composer-cli@0.20*](mailto:composer-cli@0.20)

* composer-rest-server: tiện ích để máy chủ REST hiển thị thông tin của ứng dụng dưới dạng API RESTful. Dùng lệnh:

*sudo npm install -g composer-rest-server@0.20*

* generator-hyperledger-composer: tiện ích giúp tự động. Cài đặt qua lệnh:

*sudo npm install -g* [*generator-hyperledger-composer@0.20*](mailto:generator-hyperledger-composer@0.20)

* yeoman: Công cụ tạo tạo khung ứng dụng đơn giản, từ đó em có thể sử dụng để phát triển ứng dụng riêng của mình. Công cụ này sử dụng tiện ích của generator-hyperledger-composer. Có thể coi như yeoman như là một bản thiết kế sơ bộ của một ngôi nhà, mà trong đó từng chi tiết nhỏ như mái hay cửa sổ, cửa chính... được tạo nên từ generator-hyperledger-composer. Yeoman được cài đặc thông qua lệnh:

*sudo npm install -g yo*

#### Cài đặt Hyperledger Fabric

Sử dụng các lệnh sau để cài đặt một Hyperledger Fabric chạy trên nền local máy.

* Tạo thư mục lưu (giả sử: ~/fabric-dev-servers), lấy tập tin nén chứa các công cụ cần thiết và giải nén vào thư mục lưu.

*mkdir ~/fabric-dev-servers && cd ~/fabric-dev-servers*

*curl -O* [*https://raw.githubusercontent.com/hyperledger/composer-tools/master/packages/fabric-dev-servers/fabric-dev-servers.tar.gz*](https://raw.githubusercontent.com/hyperledger/composer-tools/master/packages/fabric-dev-servers/fabric-dev-servers.tar.gz)

*tar -xvf fabric-dev-servers.tar.gz*

* Sử dụng những file scripts bên trong thư mục ~/fabric-dev-servers để tiến hành download phiên bản Hyperledger Fabric v1.2

*cd ~/fabric-dev-servers*

*export FABRIC\_VERSION=hlfv11*

*./downloadFabric.sh*

#### Cài đặt Angular

* Angular CLI: công cụ để tạo một project đơn giản tạo mã ứng dụng và thư viện và thực hiện một loạt các nhiệm vụ phát triển liên tục như kiểm tra, đóng gói và triển khai.

*npm install -g @angular/cli*

### Phát triển ứng dụng

Thương mại điện tử là một mạng lưới kinh doanh liên quan tới:

* Hàng hóa, sản phẩm như điện thoại, quần áo, rau, củ, quả...
* Bên kinh doanh: người mua, người bán.
* Thỏa thuận giữa các bên kinh doanh.

Dựa vào mạng lưới này, em sẽ xây dựng một sàn giao dịch.

#### Server extenstion

##### Model

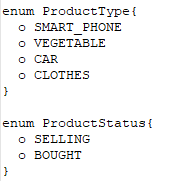
Hyperledger Composer có ngôn ngữ lập trình kiểu mô hình với tên CTO modeling languages thiết kế hướng đối tượng và khá đơn giản.

Mô hình mạng lưới ứng dụng được mô tả dưới file với định dạng .cto, được tập chung chặt chẽ với một số từ khóa: namespace, resouces, import dữ liệu từ namespace khác.

Namespace: không gian phạm vi mà mọi thứ khai báo bên trong file model này được phép sử dụng và thay đổi tùy logic nghiệp vụ. Cũng như namespace trong lập trình môn C++ thì namespace trong hyperledger composer là duy nhất.

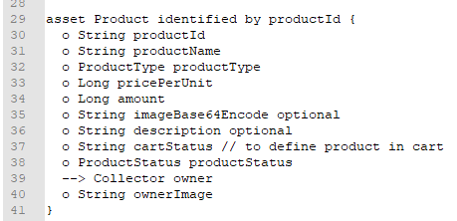
Resource: Chứa tất cả thông tin, các đối tượng trong hệ thống: assets, participants, transaction, enum type, concept.

Khai báo enum:



Hình 1: Hình mô tả các enum

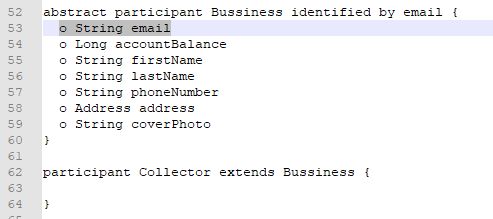
Khai báo asset:



Namespace ở đây là uet.khoenguyen.exchange.

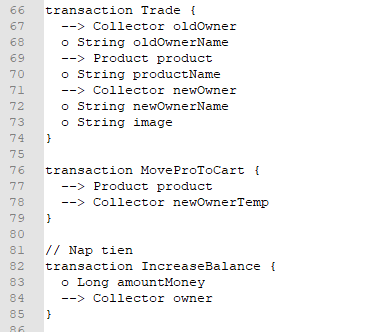
Asset Product này được định danh bởi productId, productId này là duy nhất có chức năng định danh cho mỗi sản phẩm được đăng lên hệ thống.

Khai báo participant:

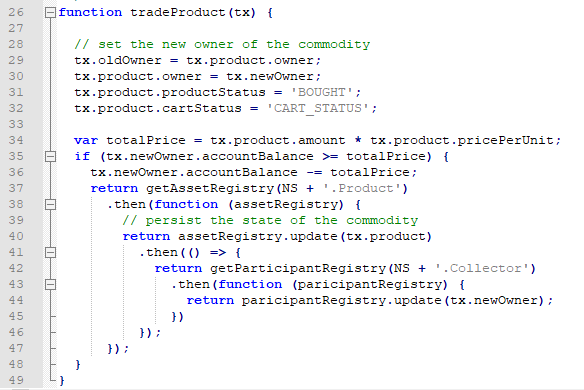


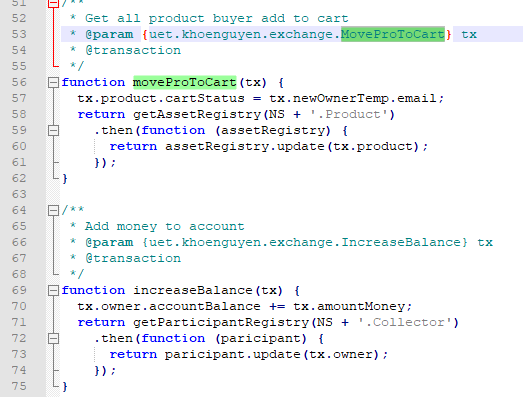
Ở đây em ứng dụng đặc tính kế thừa trong lập trình hướng đối tượng. Khai báo một participant trừu tượng Bussiness được định danh qua email. Participant kế thừa kế thừa những đặc tính từ Bussiness. Khai báo như này để đảm bảo sau này phát triển ứng dụng dễ khả rộng hơn và tối ưu code.

Khai báo các transactions:



##### Logic





##### Phân quyền

##### Truy vấn dữ liệu

#### Client

Tạo workspace vào khởi tạo ứng dụng

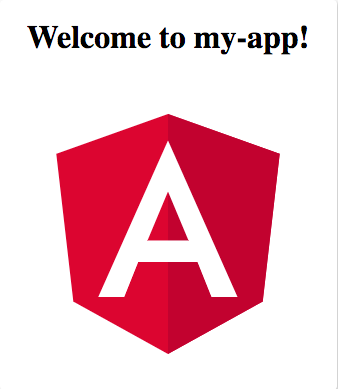
*ng new exchange-app*

Sau khi hệ thống khởi tạo xong, truy cập vào thư mục workspace (exchange-app).

Chạy thử ứng dụng lần đầu tiên qua lệnh

*ng serve*

Ứng dụng được chạy thành công với giao diện



Việc khởi tạo khung xương project đã xong. Tiếp theo, em tiến hành chỉnh sửa phát triển ứng dụng Exchange App.

Đầu tiên tạo giao diện màn hình chính khi người dùng lần đầu tiên đăng nhập ứng dụng.

<<image>>

## Quy trình khởi tạo, chạy ứng dụng

### Khởi chạy Server

#### Khởi chạy Hyperledger Fabric

cd $COMPOSER\_ROOT/fabric-dev-servers

sudo ./startFabric.sh

./downloadFabric.sh

./createPeerAdminCard.sh

Lệnh đầu tiên sẽ mất vài phút, cung cấp tất cả các thứ cần thiết vào môi trường Docker. Lệnh thứ hai để khởi chạy một Hyperledger Fabric cục bộ. Lệnh cuối cùng tạo một ID Cards cho Fabric admin – PeerAdminCard.

#### Đóng gói Hyperledger Composer server extension và kết nối tới Hyperledger Fabric

Triển khai exchange-app vào Hyperledger Fabric cục bộ, sử dụng Composer CLI và thi hành câu lệnh sau:

*cd $COMPOSER\_ROOT/Exchange/client/exchange-app*

*composer archive create -t dir*

*composer network install -a exchange@0.0.2.bna -c PeerAdmin@hlfv1 -n .*

*composer network start -n exchange -V 0.0.2 -A admin -S adminpw -c PeerAdmin@hlfv1*

*composer card import -f* [*admin@exchange.card*](mailto:admin@exchange.card)

Nén project thành một file BNA.

Lệnh thứ hai được sử dụng để cài đặt file nén BNA từ lệnh đầu tiên vào mạng Hyperledger Fabric cục bộ.

Lệnh tiếp theo, tạo một phiên bản cụ thể của mạng đã được cài đặt ở lệnh trước đó vào mạng Hyperledger Fabric. Đồng thời câu lệnh này tạo ra một thẻ admin mới. Thẻ này nén hai file connection.json và metadata.json, chứa danh tính admin và tất cả các thông tin cần thiết để kế nối tới mạng blockchain.

Ở lệnh cuối cùng, lệnh này tiến hành import thẻ admin vào ví blockchain local.

Chúng ta cũng có thể tiến hành xem các card đã được import thành công hay chưa qua lệnh:

*composer card list*

#### Khởi chạy Hyperledger composer rest server

* Server dành cho admin

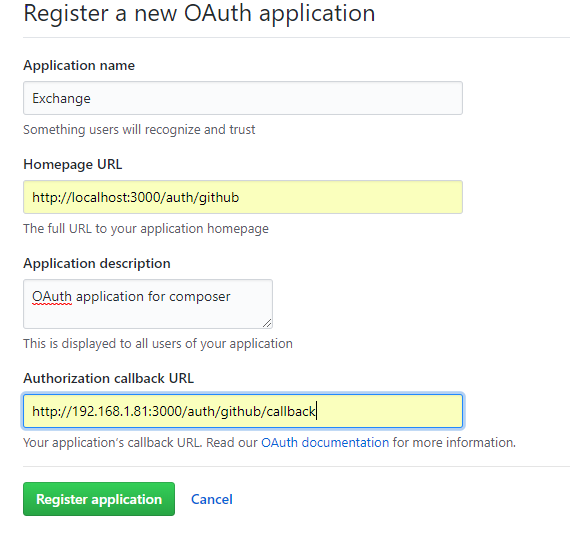
*cd HyperledgerComposers/ ExchangeApp/server/exchange/*

*composer-rest-server -c admin@exchange -p 3001 -n never*

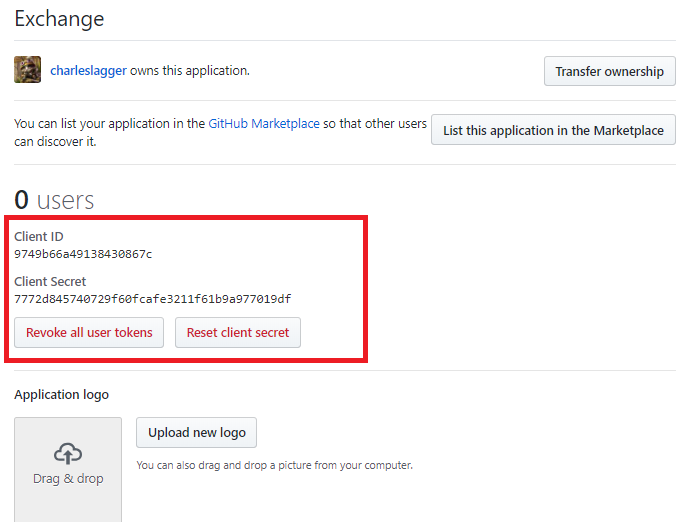
Server này được mở ở cổng 3001, sử dụng thẻ admin để kết nối tới ví và api được generate ra sẽ không bao gồm namespace như được khai báo trong model.

* Server đa người dùng

Trước tiên đăng nhập vào github, sau đó truy cập vào link <https://github.com/settings/developers>, tiến hành tạo một ứng dụng OAuth trên github điền các thông tin như hình sau



Bấm Register application và màn hình chuyển sang giao diện sau



Lưu ý vùng khoanh đỏ, copy client id và client secret để tiến hành các câu lệnh tiếp theo.

*cd HyperledgerComposers /Exchange/server/exchange/*

*export COMPOSER\_PROVIDERS='{"github":{"provider":"github","module":"passport-github","clientID":"9749b66a49138430867c","clientSecret":" 7772d845740729f60fcafe3211f61b9a977019df","authPath":"/auth/github","callbackURL":"/auth/github/callback","successRedirect":"http://localhost:4200?loggedIn=true","failureRedirect":"/"}}'*

*echo "Starting composer rest server for multiple clients at port 3000"*

*composer-rest-server -c admin@exchange -m true -n never*

Câu lệnh đầu tiên chúng ta export COMPOSER\_PROVIDERS vào Docker image, lệnh này chứa clientID và clientSecret chúng ta lấy từ github OAuth, đồng thời cung cấp các url chuyển tiếp khi xác thực github thành công hoặc thất bại.

Câu lệnh tiếp theo khởi chạy server ở chế độ đa người dùng, đầu tiên cũng sử dụng thẻ admin để kết nối tới ví và không sử dụng namespace trong các api được generate ra. Sau này khi người dùng nào đó đăng nhập vào Exchange App thì hệ thống cũng sẽ tự động generate một card tương ứng và duy nhất chứa thông tin của người dùng đó.

#### Khởi chạy client app

*cd HyperledgerComposers/Exchange/client/exchange-app*

*ng serve*

## Hiển thị kết quả

Trang chủ

Đăng nhập github

Mua hàng

Nạp tiền…

# Kết luận (10%)

## Khái quát toàn bộ nội dung đồ án

## Nhận xét, đánh giá

### Những vấn đề còn tồn tại

### Định hướng phát triển

Tài liệu tham khảo

[1] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, page 1. ACM, 2018.

[2] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, page 14. ACM, 2018.

[3] Nguyen Hoai Anh and Ao Thu Hoai. Thuong mai dien tu. page 1, 2011.

[4] Nguyen Hoai Anh and Ao Thu Hoai. Thuong mai dien tu. page 45, 2011.

[5] Christian Cachin. Architecture of the hyperledger blockchain fabric. In *Workshop on Distributed Cryptocurrencies and Consensus Ledgers*, volume 310, 2016.

[6] Vinod Kumar and Er Gagandeep Raheja. Business to business (b2b) and business to consumer (b2c) management. *International Journal of Computers & Technology*, 3(3b):447–451, 2012.

[7] Sara Rouhani, Darryl G Humphery, Luke Butterworth, Ralph Deters, and Adam D Simmons. Medichainâ€‹ tmâ€‹: A secure decentralized medical data asset management system. page 3.

[8] Sara Rouhani, Darryl G Humphery, Luke Butterworth, Ralph Deters, and Adam D Simmons. Medichainâ€‹ tmâ€‹: A secure decentralized medical data asset management system. page 4.

[9] Nguyen Thi Dieu Chi. Hoat dong thuong mai dien tu tai viet nam trong ky nguyen ky thuat so. *Tap chi Khoa hoc va Cong nghe Viet Nam*, (6A):22, 2018.

[10] Uchi Ugobame Uchibeke, Sara Hosseinzadeh Kassani, Kevin A Schneider, and Ralph Deters. Blockchain access control ecosystem for big data security. *arXiv preprint arXiv:1810.04607*, page 1, 2018.