# BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**NIÊN LUẬN**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**ỨNG DỤNG BLOCKCHAIN TRONG**

**QUẢN LÝ VĂN BẰNG**

**Sinh viên: Huỳnh Thanh Toàn**

**Mã số: B1507173**

**Khóa: K41**

**Cần Thơ, 5/2019**

# BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**NIÊN LUẬN**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**ỨNG DỤNG BLOCKCHAIN TRONG**

**QUẢN LÝ VĂN BẰNG**

**Người hướng dẫn Sinh viên thực hiện**

**TS/Ths Trần Công Án Huỳnh Thanh Toàn**

**Mã số: B1507173**

**Khóa: K41**

***Cần Thơ, XX/201Y***

**NHẬN XÉT CỦA GIÁO VIÊN**

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**LỜI CẢM ƠN**

Niên luận này được hoàn thành tại Trường Đại học Cần Thơ Trong quá trình làm báo cáo niên luân em đã nhận được rất nhiều sự giúp đỡ để hoàn tất nó.

Trước tiên em xin gửi lời cảm ơn chân thành thầy Trần Công Án đã tận tình hướng dẫn, truyền đạt kiến thức, kinh nghiệm cho em trong suốt quá trình thực hiện luận văn tốt nghiệp này.

Xin gửi lời cảm ơn đến quý thầy cô Khoa Công Nghệ Thông Tin và Truyền Thông, Trường Đại học Cần Thơ, những người đã truyền đạt kiến thức quý báu cho em suốt trong thời gian học tập vừa qua. Vì kiến thức còn hạn chế và còn thiếu kinh nghiệm thực tiễn nên báo cáo của em còn nhiều thiếu xót, em rất mong nhận được sự góp ý của thầy để báo cáo được hoàn thiện hơn.

Một lần nữa, xin chân thành cảm ơn!

Cần Thơ, ngày 10, tháng 5, năm 2019

Người viết

Huỳnh Thanh Toàn

**TÓM TẮT**

Năm 2018, cả thế giới phát sốt vì sự biến động giá của BitCoin, mọi người đổ xô đi mua Bitcoin như một hình thức đầu tư. Nhưng ít người biết rằng công nghệ đứng sau cơn sốt ấy chính là Blockchain. Bên cạnh được ứng dụng làm đồng tiền điện tử, Blockchain còn có rất nhiều ứng dụng hữu ích cho cuộc sống, nhất là khi quốc gia đang bước vào cuộc cách mạng 4.0.

Công nghệ Blockchain đang phát triển ngày một nhanh, Blockchain đang nổi lên như một trong những công nghệ nền tảng thay đổi cách chúng ta trao đổi, lưu trữ thông tin thông qua internet.

Blockchain có thể ứng dựng vào nhiều lĩnh vực khác nhau như y tế, tài chính-ngân hàng, nông nghiệp, giáo dục. Đơn cử, tron ngành giáo dục năm qua nổi lên nhiều vụ việc gian lận, nếu áp dụng công nghệ Blockchain vào việc quản lý các kỳ thi, các cơ sở dữ liệu bằng cấp thì sẽ hạn chế đến mức tối đa những vấn đề trên.

Nhưng hầu như các ứng dụng đấy chưa có nhiều khả năng phát triển ở Việt Nam. Một trong những lý do lớn nhất là do cơ sở hạ tầng ban đầu của nước ta vẫn còn chưa đáp ứng được. Số lương nguồn nhân lực chất lượng cao vẫn chưa đủ đáp ứng.

Để mô phỏng đơn giản lại quá trình vận hành của blockchain trong việc quản lý bằng cấp, em đã chọn và xây dựng đề tài “Ứng dụng Blockchain trong việc quản lý văn bằng”.

PHẦN GIỚI THIỆU

Đặt vấn đề

Thực trạng bằng giả hiện nay đang là một trong những vấn đề nhức nhối của xã hội. Nhà tuyển dụng ngày càng lo sợ về việc thuê phải người không có tay nghề thực, Nhà trường càng lo lắng hơn khi uy tính bị đem ra mua bán.

Do nhu cầu trên thị trường rất lớn, nên các dịch vụ mua bán bằng ngày càng hoạt động công khai hơn. Tùy vào loại trường, loại bằng cấp mà giá thành của nó khác nhau.

Để giải quyết ván đề trên, em đề xuất việc sử dụng blockchain trong việc quản lý các văn bằng tốt nghiệp. Bằng việc liên kết với các trường, hệ thống sẽ ghi lại tất cả thông tin về sinh viên mới tốt nghiệp vào Blockchain và sẽ không bao giờ bị thay đổi. Mọi văn bằng mới đươc thêm vào đều được ghi danh xưng người tạo, nên một người không có thẩm quyền sẽ không có quyền ghi vào blockchain.

Những nghiên cứu liên quan

Trong lĩnh vực y tế

Nebula Genomics đã ứng dụng blockchain trong việc lưu trữ gen người. Sách trắng về gen của Nebula hứa hẹn sẽ tạo ra một thị trường dữ liệu gen hang tỷ đô la.

Doc.AI một start up đã ứng dụng blockchain và AI trong việc lưu trữ dữ liệu sức khỏe bênh nhận và Giao tiếp với người bện đặt lịch khám một cách tự động hóa.

Trong lĩnh vực tài chính

Hiện tại có hơn 200 ngân hàng và tổ chức tài chính trên thế giới ứng dụng blockchain

Trong lĩnh vực Giáo dục

Blockcerts đã xây dựng một nền tảng mã nguồn mở để lưu trữ văn bằng, hơn 600/2018 sinh viên tốt nghiệp MIT đã chọn nhận một tấm bằng điện tử trên hệ thống blockchain này.

GilGamesh là một nền tảng chia sẻ tri thức dựa trên Ethereu smart contact, họ dung tiền ảo để giao dịch như việc mua sách điện tử của sinh viên.

Mục tiêu đề tài

Tìm hiểu cách vận hành của Ethereum Blockchain.

Tìm hiểu cách hoạt động của Smart Contact Ethereum.

Xây dựng UI đơn giản mô phỏng việc giao tiếp với hệ thống.

Đáp ứng nhu cầu bảo mật việc truy xuất dữ liệu từ Blockchain

Đối tượng và phạm vi nghiên cứu

Đối tượng: Công nghệ Blockchain, Ethereum, Smart Contact, Truffle, MetaMask, Infura, …

Phạm vi: Phạm vi của đề tài được giới hạn trong lĩnh vực quản lý bằng đại học.

Phương pháp nghiên cứu

Tìm hiểu các thông tin của một văn bằng.

Nghiên cứu cách viết smart contact.

Tìm hiểu cách tạo mạng Ethereum.

Tìm hiểu cách Web giao tiếp với mạng Ethereum

Nội dung nghiên cứu

Server: thiết lập mạng Ethereum, triển khai Smart contact lên mạng Ropsten Test Net

Client: Thực hiện kết nối vào mạng Ethereum Ropsten Test Net, tạo giao dịch, quản lý nhật ký truy xuất.

PHẦN NỘI DUNG

CHƯƠNG 1

GIỚI THIỆU CÔNG NGHỆ

Blockchain

Blockchain (chuỗi khối) là một cơ sở dữ liệu phân cấp lưu trữ thông tin trongh các khối thông tin được liên kết với nhau dựa trên chuỗi mã hóa khối trước đó và mở rộng theo thời gian. Mỗi khối đều được liên kết khối liền kề trước đó, kèm theo một chuỗi mã hóa về thời gian và thông tin giao dịch. Khi một khối được tao hay một dữ liệu nào đó được đưa vào thì không thể nào điều chỉnh được, kể cả việc xóa bỏ khối đó cũng không thể.

Blockchain có khả năng chịu lỗi cao bằng việc sử dụng hệ thống tính toán phân cấp. Vì vậy, Blockchain phù hợp để lưu lại những sự kiện về hồ sơ y tế, giáo dục, giao dịch tài chính, hồ sơ công dân, …Việc này giúp giảm bớt tỉ lệ thiệt hại về việc dữ liệu bị thay đổi.

Ethereum

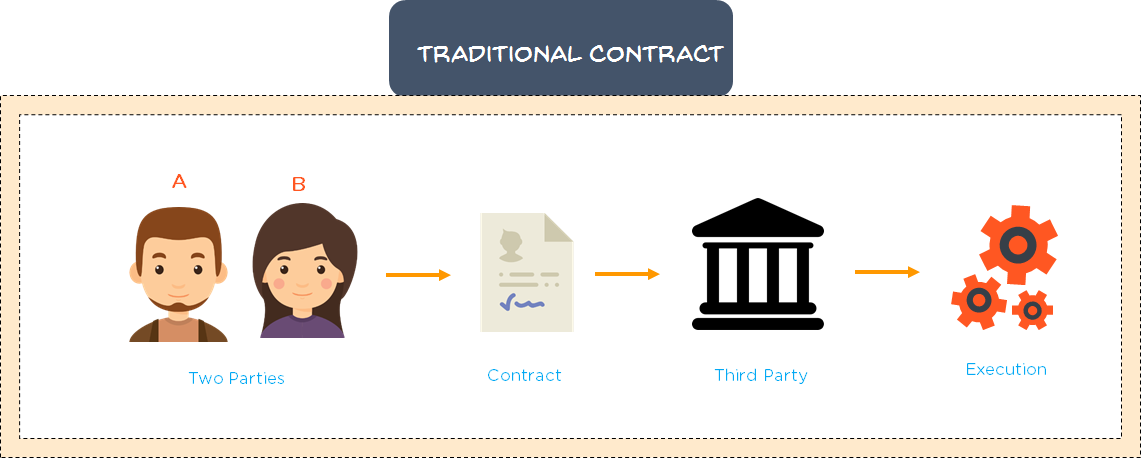
Ethereum là một nền tảng điện toán có tính chất phân tán, công cộng, mã nguồn mở dựa trên công nghệ Blockchain. Thuận lợi cho các thỏa thuận hợp đồng trực tuyến. Nó bao gồm một máy ảo hoàn toàn Turing – Ethereum Virtual Machine (EVM), có thể sử dụng bằng cách sử dụng một mạng lưới máy Ethereum, Ethereum cũng cung cấp một loại tiền ảo gọi là “Ether”, có thể chuyển tiền giữa các tài khoản và sử dụng để trả công cho các Miner (thợ đào).

Ethereum còn là nền tảng ứng dụng hữu ích và đã tạo ra được một hệ sinh thái tài chính phân tán cho riêng mình.

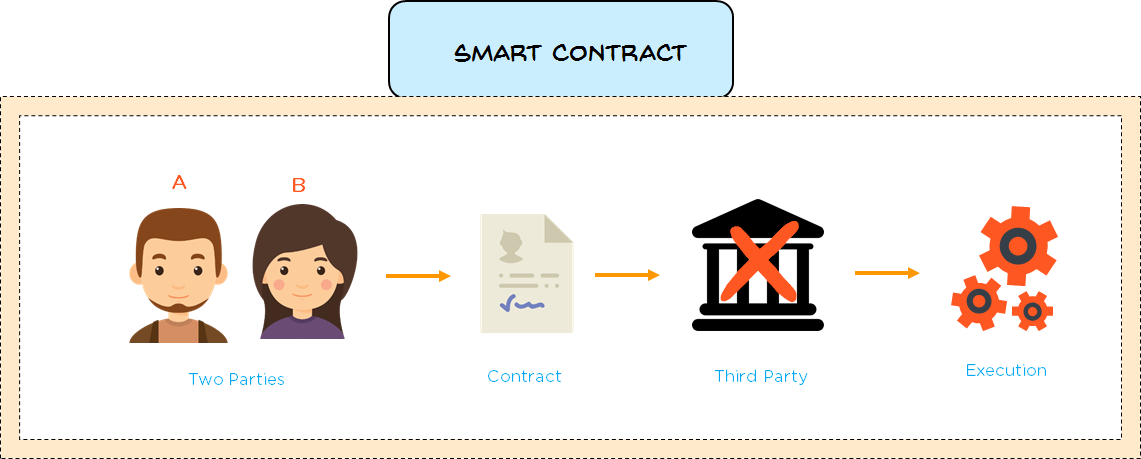
Có thể xây dựng một “smart contract” bằng cách tạo ra một giao dịch không có địa chỉ đích và chứa một đoạn code để chỉ ra nó phải làm gì. Các smart contract này sau đó sẽ được xử lý bởi các thợ mỏ đào Ethereum và được thêm vào Blockchain, lưu trữ vĩnh viễn trong sổ cái và được phân phối cho tất cả máy tính trong hệ thống. Nếu ai đó tương tác với hợp đồng này (ví dụ chuyển một số tiền vào địa chỉ công khai của người tạo) thì hợp đồng sẽ được thực thi và các Block sẽ được khởi tạo và thêm vào hệ thống Blockchain.

Smart Contract

Smart Contract (hợp đồng thông minh là một thuật ngữ mô tả khả năng tự đưa ra các điều khoản và thực thi thoả thuận của hệ thống máy tính bằng cách sử dụng Blockchain. Toàn bộ quá trình của Smart Contract được thực hiện tự động và không có sự can thiệp từ bên ngoài. Hai bên chủ thể hợp đồng không cần thông qua bên thứ ba để xác nhận hợp đồng, vì hợp đồng luôn luôn độc lập so với cả hai bên. Các điều khoản của Smart Contract tương đương với một hợp đồng pháp lý và được ghi lại dưới ngôn ngữ của máy tính.



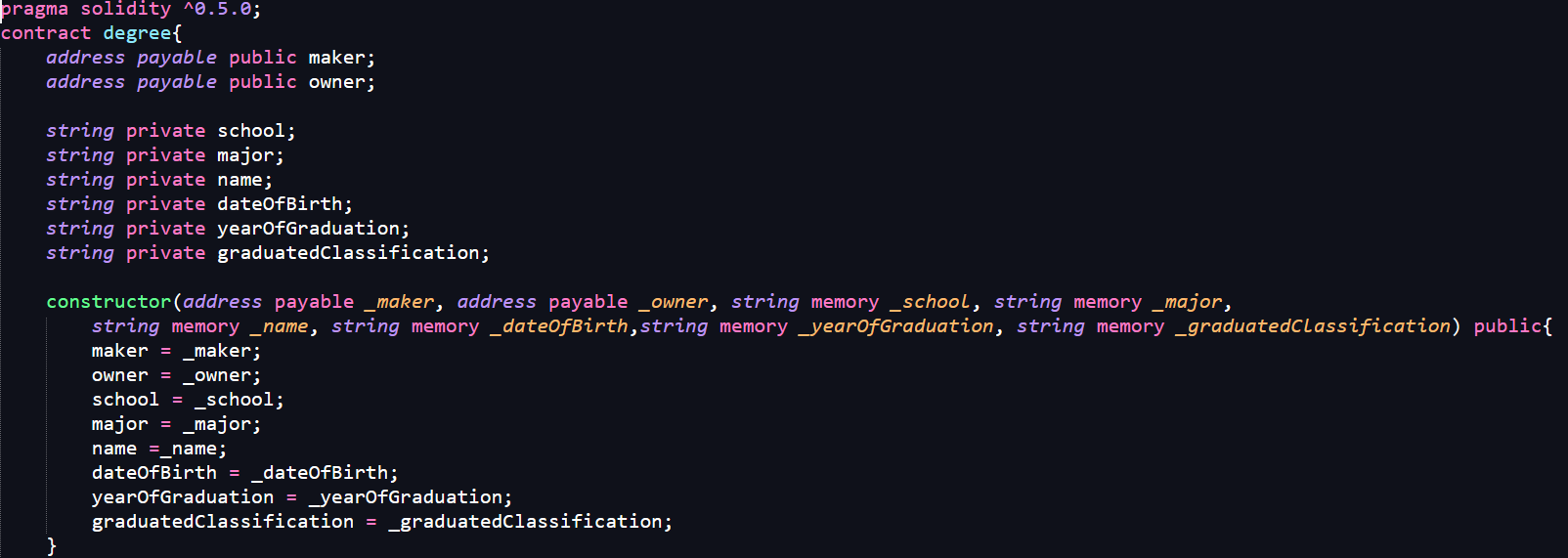
Hình. Hợp đồng truyền thống



Hình. Hợp đồng thông minh

Trong Ethereum, người lập trình được quyền tạo ra những Smart Contract tùy ý, nó sẽ được thực thi một cách tự động khi những điều kiện của nó được đáp ứng. Nguyên lý hoạt động của nó được dựa trên một nguyên lý đơn giản “nếu…thì”. Một ví dụ đơn giản về máy bán hàng tự động, nếu bạn cho tiền vào máy và số tiền đó lớn hơn hoặc bằng giá của sản phẩm thì máy sẽ giao sản phẩm cho bạn.

Smart Contract của Ethereum được viết bằng ngôn ngữ Solidity, một loại ngôn ngữ hướng đối tượng dành riêng cho nó.

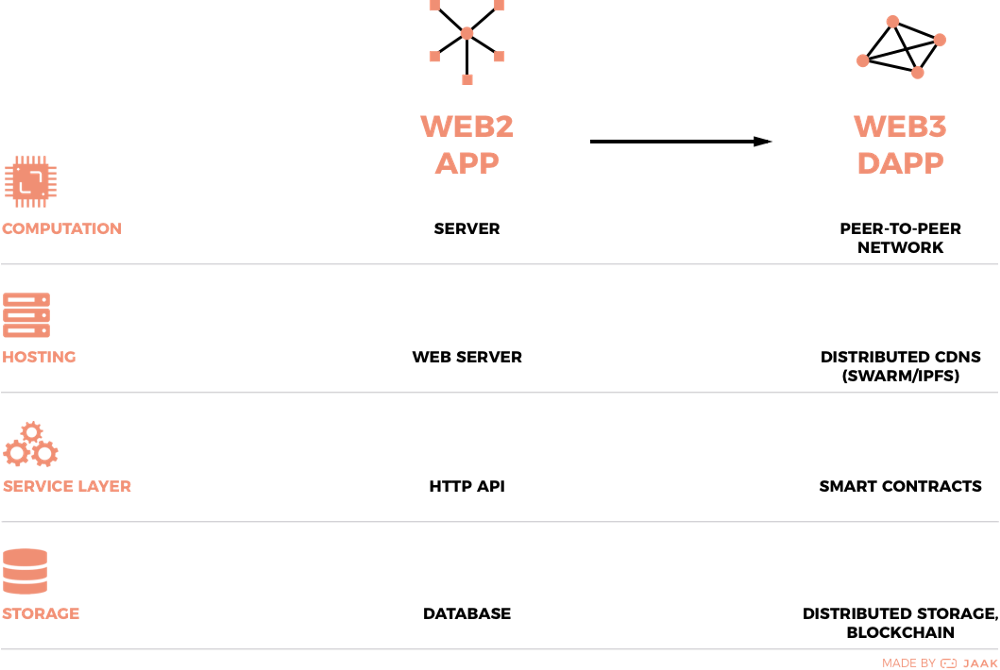


Hình. Ngôn ngữ solidity

Smart contract – Hợp đồng thông minh đã tạo nên sự khác biệt cho Ethereum với việc ứng dụng công nghệ Blockchain. Các nhà phát triển và sáng tạo đang tạo nên một nền tảng có bước đột phá lớn hoặc với tên gọi khác là “Web 3.0”.

Web3

Web3 là thế hệ tiếp theo của world wide web. Nó được đề xuất bởi Ethereum Ecosystem. Nói một cách đơn giản web3 là web2 với server và database phân tán. Dapp là decentralizes app, dapp không có server. “Back end” code được chạy trên Mạng ngang hàng (peer to peer), “front end” code được chạy trên CDN (Content Delivery Network) một mạng chứa các bản sao của nội dung.



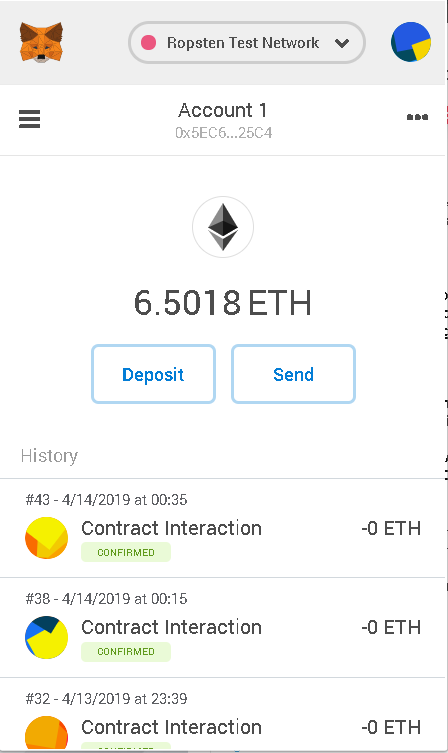
Hình. ứng dụng phân tán

Truffle

Truffle một môi trường phát triển, thử nghiệm cho Ethereum, giúp cho việc xây dựng và triển khai các Smart Contract của Ethereum trở nên dễ dàng hơn. Truffle có cơ chế hỗ trợ quản lý các mạng triển khai đơn giản bao gồm cả public và private network.

Metamask

Metamask là một plugin dùng để quản lý các tài khoản tiền ảo cho bạn, tránh các trường hợp bị đánh cắp. Nó cho phép bạn truy cập vào ethereum dapp trong trình duyệt nhờ vào private key tài khoản của bạn mà không cần khởi động toàn bộ node ethereum.



Hình. Metamask

Infura

Infura là một nền tảng công cấp các công cụ kết nối các ứng dụng phi tập trung (dapp) với Ethereum network thông qua các API. Bạn có thể dùng Infura để kết nối tới hầu hết các network lớn như Ropsten, Rinkeby, Kovan testnet.

CHƯƠNG 2

CÀI ĐẶT

Cài đặt hệ thống

Server

Hệ thống đảm nhiệm chức năng server của Dapp cần những thành phần:

Solidity v0.5.0

Truffle v5.0.0

Infura

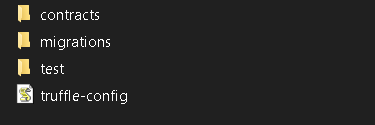
Để cài đặt môi trường Truffle, ta phải cài đặt Nodejs và gói NPM. Sau đó cài truffle bằng gói NPM như sau:



Khởi tạo thư mục làm việc với truffle:



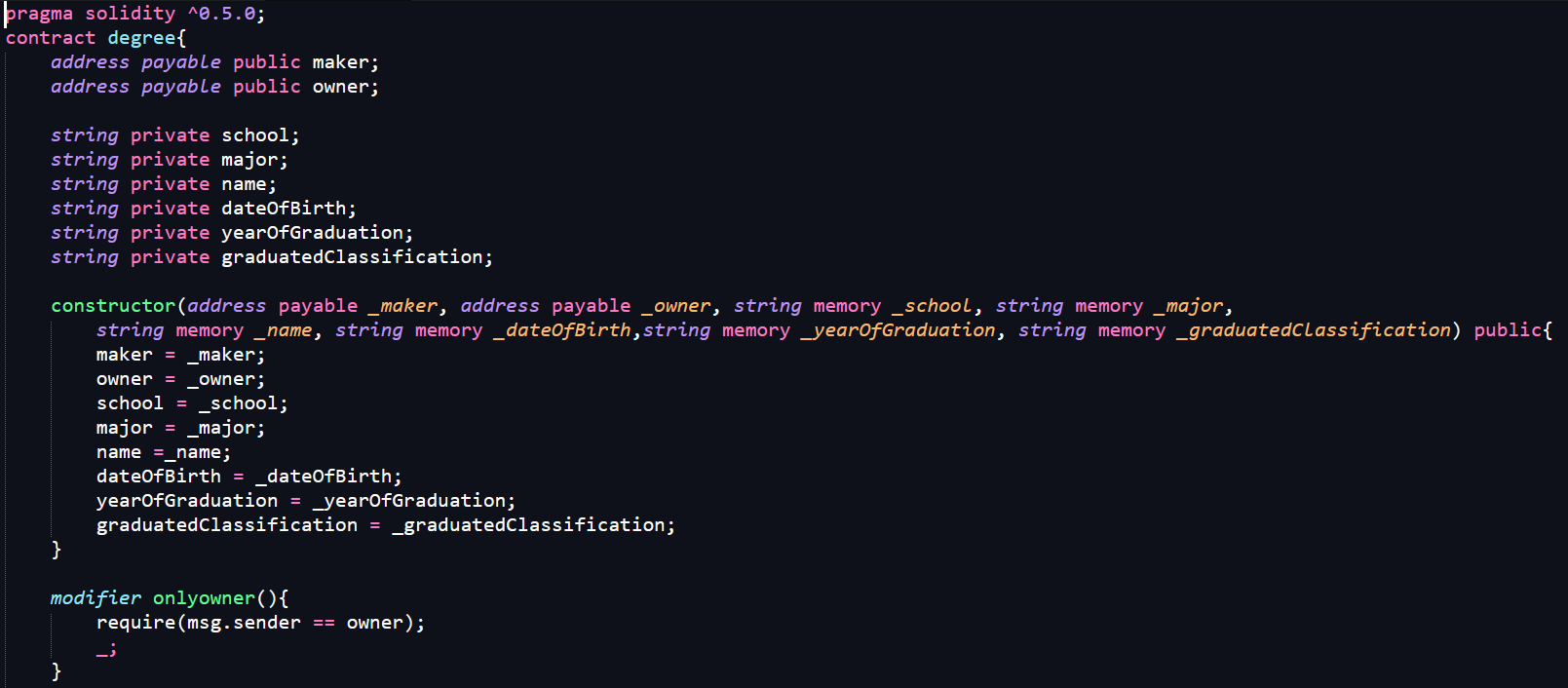
Cấu trúc thư mục truffle:



Thư mục contracts sẽ chứa các smart contract cần được triển khai. Thư mục migrations chứa các file java script để hỗ trợ quá trình triển khai smart contract. File truffle-config.js sẽ chứa địa chỉ, các thông số của mạng blockchain đích.

Smart contract

Để xây dựng Dapp này, em đã tạo ra 2 smart contract: degree.sol, createDegree.sol. Các smart contract được viết bằng ngôn ngữ solidity, theo hướng lập trình hướng đối tượng.



Hình. degree

Degree.sol là smart contract chứa cấu trúc của một văn bằng bao gồm các trường: maker chứa địa chỉ tài khoản người tạo ra văn bằng đó, owner sẽ chứa địa chỉ tài khoản của chủ nhân văn bằng. school, major, name, dateofbirth, yearofgraduation, graduatedClassification sẽ chứa các thông tin cơ bản về văn bằng đó.



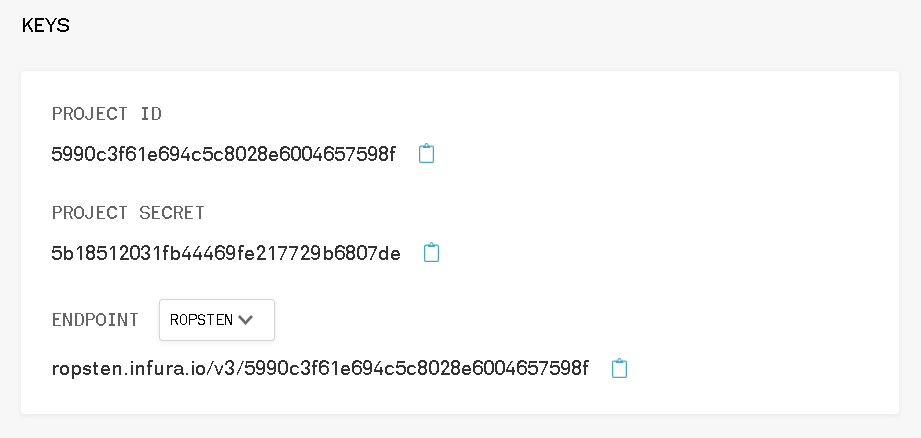
Hình. Createdegree

Trong createDegree.sol sẽ chứa các quá trình chính của phần mềm như việc tạo văn bằng, ghi log, lưu trữ văn bằng. Struct degreestruct sẽ chứa cấu trúc của một văn bằng giống như contract “degree” trong degree.sol, việc tạo struct như vậy để phục vụ mục đích dễ dàng thao tác truy xuất các thông tin của văn bằng hơn. Mapping là một dạng cấu trúc của solidity giống như mảng động.

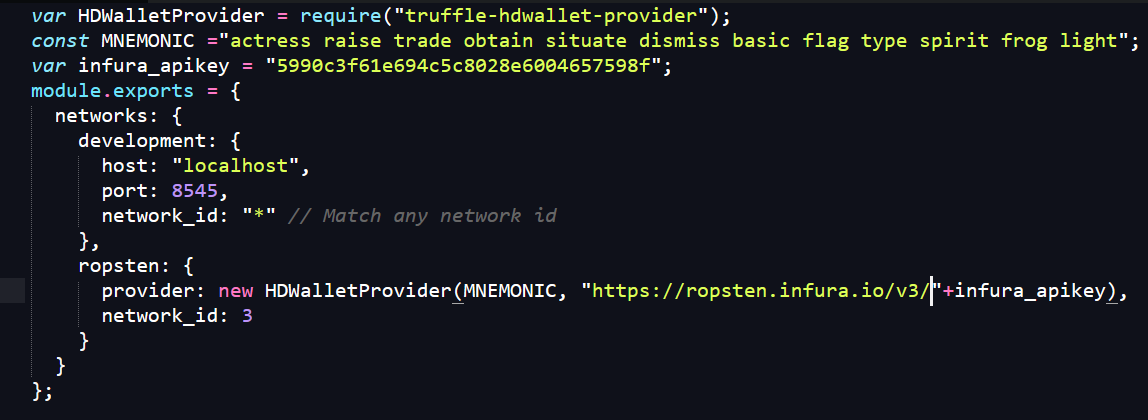
Mapping “degrees” sẽ chứa các “degreeStruct” văn bằng đã được tạo. Mapping degreeAddress chúa thông tin về địa chỉ block mà các contract “degree” được lưu, từ địa chỉ trên ta có thể triển khai một degree nào đó thành một chain riêng từ đó có thể phát triển thêm một số tín năng nâng cao như trả phí khởi văn bằng, …

DegreeCount chứa thông tin về số văn bằng đã được lưu trên hệ thống. createLog chứ bản ghi các hoạt động liên quan tới việc thêm văn bằng mới.

Tạo dự án trên Infura, chọn blockchain network là Ropsten Test Network. Lấy địa chỉ Endpoint có được từ dự án trên Infura, điều chỉnh lại file truffle-config.js và source-api cho client.



Hình. Infura-config



Hình. Truffle-config

Ta có thể triển khai smart contract lên server blockchain nội bộ của Truffle, nhưng có một bất tiện là mọi dữ liệu ta đã nhập vào lần trước để test sẽ bị xóa hết mỗi lần ta tắt server. Để thuận tiện và gần sát với thuật tế nhất, em sẽ chọn làm việc mạng Ropsten Test Net. Những giá trị như Gas limit, Gas price sẽ là mặc định.

Triển khai contract lên blockchain network.



Client

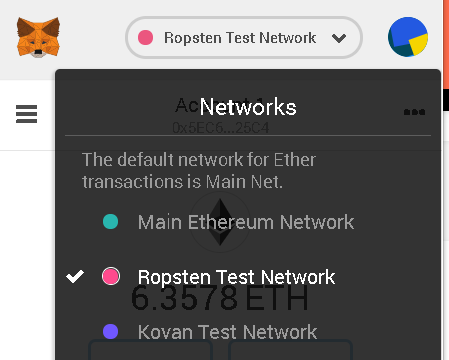
Để kết nối tới Server blockchain, một client phải có những thành phần sau:

Webpack

Metamask

Nodejs

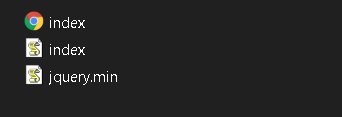
Cài đặt infura và thiết đặt network là Ropsten.



Hình. Metamask-choose-network

Ta sử dụng Web3 và một số công cụ khác trong gói Webpack để xây dựng phương thức kết nối client với Blockchain Server. Gói Webpack cũng được cung cấp bởi NPM.

Thư mục làm việc chính của Webpack:

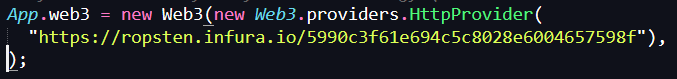


Hình. App-folder

File index.html là trang chủ của phần mềm, jquery.min.js là thư viện được dùng trong việc file index.js.

Quá trình kết nối

Để kết nối client với server trên ta sử dụng thư viện Web3



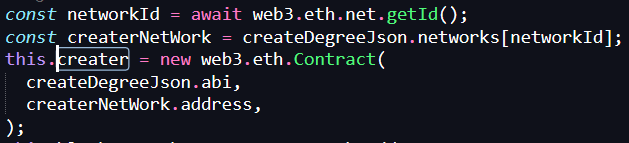
Hình. Client-connect

Địa chỉ trên là địa chỉ dự án của infura mà ta đã tạo ở trên. Câu lệnh trên gửi yêu cầu kết nối tới server blockchain và nhận về một api. Để có thể truy xuất thông tin từ api trên, ta còn gói json sinh ra từ việc thực thi các smart contract.



Hình. Json-file

Để tạo ra một đối tượng có thể thao tác truy xuất trên nó, ta phải kết hợp hai đối tượng ở trên

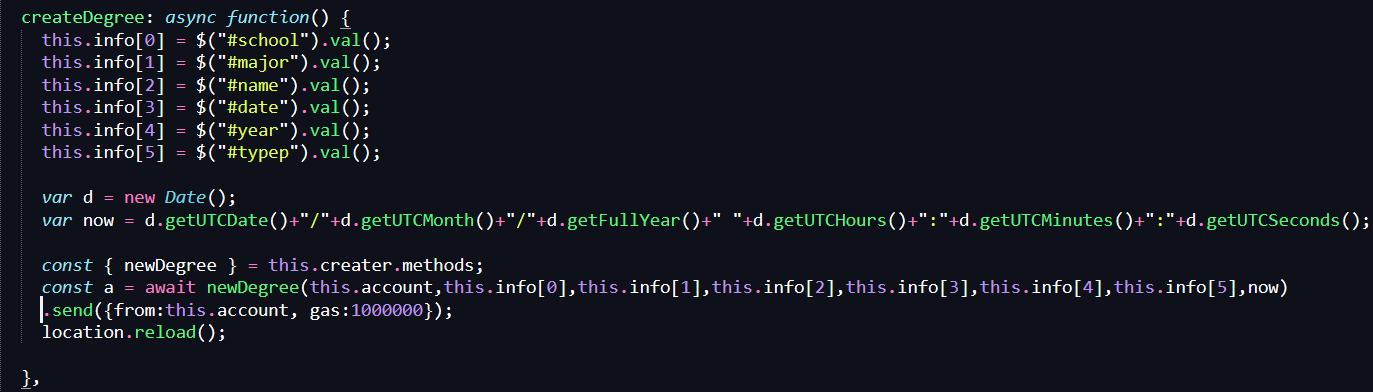


Hình. Merger-json-api

Bây giờ ta có thể gọi các phương thức, thủ tục đã khai báo trong smart contract với đối tượng chính “creater”.

Transaction

Khi tạo một giao dịch, một hoạt động nào đó có phát sinh block, ví dụ như việc tạo ra một văn bằng mới trong phần mềm này. Việc đó sẽ phải trả một mức phí nhất định, mức phí ít nhất phụ thuộc vào dung lượng nội dung mà bạn đang muốn ghi vào block mới. Mức phí này được hệ thống chi trả cho các “thợ đào”, Miner tham gia vào việc ghi block đó vào hệ thống Blockchain Network. Trong phần mềm này khi bạn click vào nút Create trong mục tạo văn bằng mới, hệ thống sẽ gọi ví Metamask lên và xác nhận lại giao dịch một lần nữa với việc hiển thị các thông tin của giao dịch sắp diễn ra cũng như mức phí mà bạn phải trả, khi click vào Confirm đồng nghĩa với việc bạn đã đồng ý chi trả cho số phí đó và sau khoảng 5-10s giao dịch sẽ hoàn tất. Thời gian đó là thời gian để các Miner tìm ra các thông số phù hợp với Blockchain để thực hiện ghi block đó một cách hợp lệ.



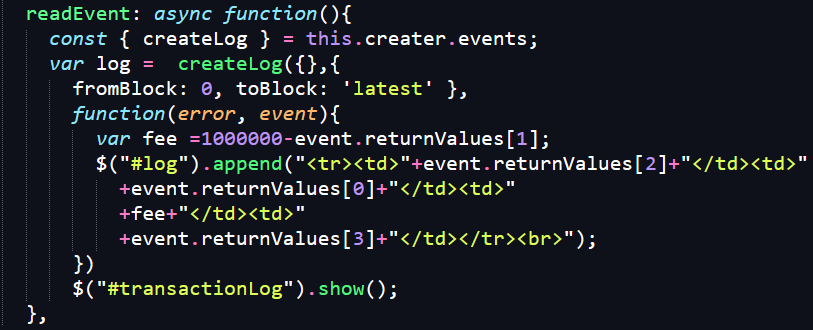
Hình. Create-function

Khi gọi hàm newDegree của Smart Contract và sau đó là hàm send() để thực hiện việc strả phí, với tham số phụ là from và gas, ta đã quy định tài khoản trả phí là tài khoản đang thực hiện việc thêm mới này và mức phí tối đa là 1000000 gas tương đương khoảng 0.1 ETH.

Trong Solidity có hỗ trợ việc tạo một nhật ký riêng (log), việc này sẽ giúp chúng ta dễ dàng quản lý được các nội dung mà ta cần ghi lại sau mỗi giao dịch. ở phần mềm này, em đã tạo ra một log để lưu lại thông tin của việc tạo một văn bằng mới như tên, mức phí, thời gian tạo, địa chỉ block chứ và người đã tạo.

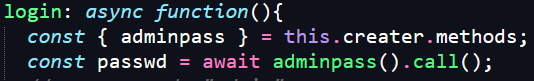


Gọi hàm để truy xuất event đã tạo ở trên:



Hình. Read-events

Đối với các hàm, phương thức không có tạo ra một transaction mới hay block mới thì ta dùng hàm call() để gọi hàm đó từ smart contract. Dưới đây là ví dụ về việc lấy password để đăng nhập chức năng Admin



Hình. Login-pass

Để có thể call() ta phải khai báo một biến truy cập tới đối tượng chứa phương thức của đối tương chính “creater”.

Mô tả giao diện

Giao diện chính (Guess) của phần mềm

Phần mềm cung cấp cho Guess một bảng gồm tất cả các văn bằng đang lưu trữ trên hệ thống, cùng với đó là chức năng tìm kiếm thuận tiện.

Giao diện quản lý (Admin) của phần mềm

Admin có thể biết được địa chỉ của mạng Blockchain hiện tại, địa chỉ của tài khoản-ví đang được sử dụng, cũng như số văn bằng đang được lưu trữ trên hệ thống của mình.

Admin có quyền đọc log về việc tạo văn bằng mới của tất cả văn bằng đang tồn tại trên hệ thống, ngoài ra cũng có thể tìm kiếm một hay nhiều văn bằng nào đó như Guess.

Và Chức năng chính của admin là tạo ra một văn bằng mới.