TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo

Description automatically generated

**BÁO CÁO CUỐI KỲ MÔN BLOCK-CHAIN**

*Người thực hiện:* **TRẦN MINH TRÍ - 52000815**

**TRẦN VĂN THẢO - 52000805**

**TRẦN QUỐC VINH - 52000823**

Lớp: **200503401**

Khóa: **24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo

Description automatically generated

**BÁO CÁO CUỐI KỲ MÔN BLOCK-CHAIN**

*Người thực hiện:* **TRẦN MINH TRÍ - 52000815**

**TRẦN VĂN THẢO - 52000805**

**TRẦN QUỐC VINH - 52000823**

Lớp: **200503401**

Khóa: **24**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Nhóm xin chân thành cảm ơn thầy Phạm Thái Kỳ Trung đã xây dựng nên một khóa học Blockchain giúp tụi em tìm hiểu, nắm bắt được công nghệ chuỗi khối là như thế nào. Bên cạnh đó, tạo điều kiện cho chúng em có cơ hội được làm việc cùng nhau. Thông qua đồ án cuối kỳ này, cá nhân mỗi thành viên đều được tiếp cận những kiến thức mới liên quan tới quá trình tạo một mạng blockchain, cách kết nối và giao tiếp với chúng với database truyền thống, được học hỏi lẫn nhau trong quá trình xây dựng application.

*TP. Hồ Chí Minh, ngày 29 tháng 12 năm 2023*

*Tác giả*

*Trần Minh Trí*

*Trần Văn Thảo*

*Trần Quốc Vinh*

BÁO CÁO ĐƯỢC HOÀN THÀNH

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Khóa luận/Đồ án tốt nghiệp còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung Báo cáo cuối kỳ của mình**. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 29 tháng 12 năm 2023*

*Tác giả*

*Trần Minh Trí*

*Trần Văn Thảo*

*Trần Quốc Vinh*

**TÓM TẮT**

Bài báo cáo được chia làm 4 chương. Chương 1 là phần tổng quan về đề tài, xoay quanh các vấn đề như các thách thức trong lĩnh vực buôn bán bất động sản truyền thống và giới thiệu về nền tảng Lofty. Chương 2 sẽ là phần phân tích thiết kế hệ thống dựa trên mô hình Water Fall gồm các sơ đồ và đặc tả usecase, sơ đồ lớp, sơ đồ tuần tự, sơ đồ hoạt động, ERD, mô hình quan hệ. Chương 3 là phần hiện thực hệ thống gồm các công nghệ sử dụng, cách setup và chạy hệ thống, giao diện và demo các chức năng chính. Chương 4 nói về hướng phát triển và ý nghĩa thực tiễn của đề tài. Dựa trên những gì đã học nhóm chúng em đã cố gắng hết sức để áp dụng hết những gì đã học ở môn này kết hợp kinh nghiệm ở những môn khác có liên quan để tạo nên hệ thống này. Tuy nhiên, dự án của nhóm vẫn có thể phát sinh một số lỗi chưa tìm thấy, mong thầy thông cảm. Nhóm xin chân thành cảm ơn.

**MỤC LỤC**

[**TÓM TẮT v**](#_Toc27245)

[**DANH MỤC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ viii**](#_Toc25024)

[**1 CHƯƠNG 1 - TỔNG QUAN VỀ ĐỀ TÀI 1**](#_Toc13903)

[1.1 Các thách thức trong lĩnh vực buôn bán bất động sản (bđs) truyền thống 1](#_Toc29885)

[1.2 Nội dung đề tài và giới thiệu nền tảng Lofty 2](#_Toc532)

[**2 CHƯƠNG 2 - PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 3**](#_Toc18780)

[2.1 Đặc tả yêu cầu chức năng 3](#_Toc2569)

[2.2 Đặc tả yêu cầu phi chức năng 4](#_Toc5924)

[2.3 Lược đồ và đặc tả chức năng (2) 5](#_Toc2654)

[2.4 Lược đồ và đặc tả quan hệ thực thể (ER diagram/specification) 19](#_Toc2365)

[**3 CHƯƠNG 3 - HIỆN THỰC HỆ THỐNG 20**](#_Toc19052)

[3.1 Các công nghệ sử dụng trong đề tài 20](#_Toc14116)

[3.2 Cách setup và chạy hệ thống 21](#_Toc25378)

[3.3 Giao diện và demo các chức năng chính 23](#_Toc16363)

[**4 CHƯƠNG 4 - KẾT LUẬN 23**](#_Toc8628)

[4.1 Hướng phát triển 23](#_Toc1179)

[4.2 Ý nghĩa thực tiễn 24](#_Toc12559)

[**BẢNG PHÂN CHIA CÔNG VIỆC 25**](#_Toc8812)

[**TÀI LIỆU THAM KHẢO 26**](#_Toc18330)

**DANH MỤC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ**

[Bảng 1 : Yêu cầu chức năng phía User 3](#_Toc26852)

[Bảng 2 : Yêu cầu chức năng phía Admin 4](#_Toc23755)

[Bảng 3 : Usecase Verify Property 7](#_Toc31523)

[Bảng 4 : Listing Property 9](#_Toc26600)

[Bảng 5 : User Manage Property 11](#_Toc7807)

[Bảng 6 : Use case Listing Property 14](#_Toc2308)

[Bảng 7 : Use case buy or sell token 16](#_Toc13131)

[Bảng 8 : Use case withraw rental payment 17](#_Toc32418)

[Bảng 9 : Use case view and cancel order 19](#_Toc9423)

[Hình 1 : Lược đồ Usecase Admin verify property 5](#_Toc13620)

[Hình 2 : Admin listing property 8](#_Toc31968)

[Hình 3 : User Manage Property 10](#_Toc21851)

[Hình 4 : User Request Listing Property 11](#_Toc18247)

[Hình 5 : Usecase buy or sell token 14](#_Toc25935)

[Hình 6 : Use case Withdraw Rental Payment 16](#_Toc25752)

[Hình 7 : Use case view or cancel orders 18](#_Toc14777)

# **CHƯƠNG 1 - TỔNG QUAN VỀ ĐỀ TÀI**

## **Các thách thức trong lĩnh vực buôn bán bất động sản (bđs) truyền thống**

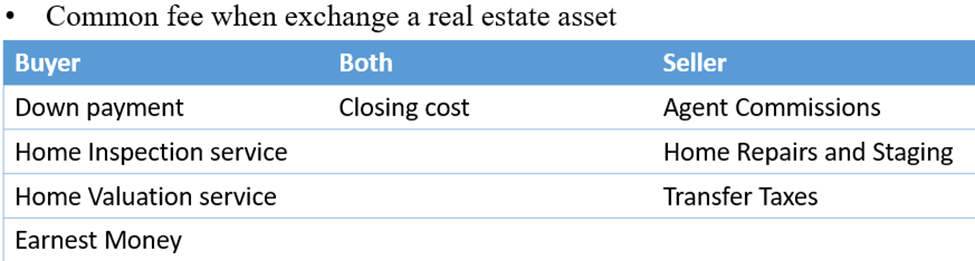
Bất động sản Việt Nam đang trải qua giai đoạn phát triển mạnh mẽ, đặc biệt là tại các thành phố lớn như Hà Nội và TP.Hồ Chí Minh. Giá nhà và đất tăng đột biến, tạo ra cơ hội đầu tư và tham gia vào thị trường cho nhà đầu tư trong và ngoài nước.

Tuy nhiên các giao dịch trong bất động sản truyền thống cũng thường gặp phải nhiều thách thức như:

* **Số lượng bên thứ ba cao**: BĐS là loại tài sản có giá trị cao và gắn với nhiều vấn đề pháp lý. Do đó các giao dịch liên quan tới BĐS cũng yêu cầu nhiều bên thứ ba tham gia
  + Ngân hàng: hỗ trợ các khoản vay
  + Luật sư/Công chứng viên: xử lý các loại giấy tờ pháp lý
  + Trung tâm môi giới: hỗ trợ người mua và bán trong quá trình giao dịch
  + Các dịch vụ thẩm định/định giá nhà
  + Cơ quan chính phủ

Þ Mỗi một bên thứ ba đều yêu cầu lượng phí nhất định, dẫn đến chi phí giao dịch cuối cùng cho cả người mua lẫn bán là rất cao và tốn kém thời gian do phụ thuộc vào bên thứ ba

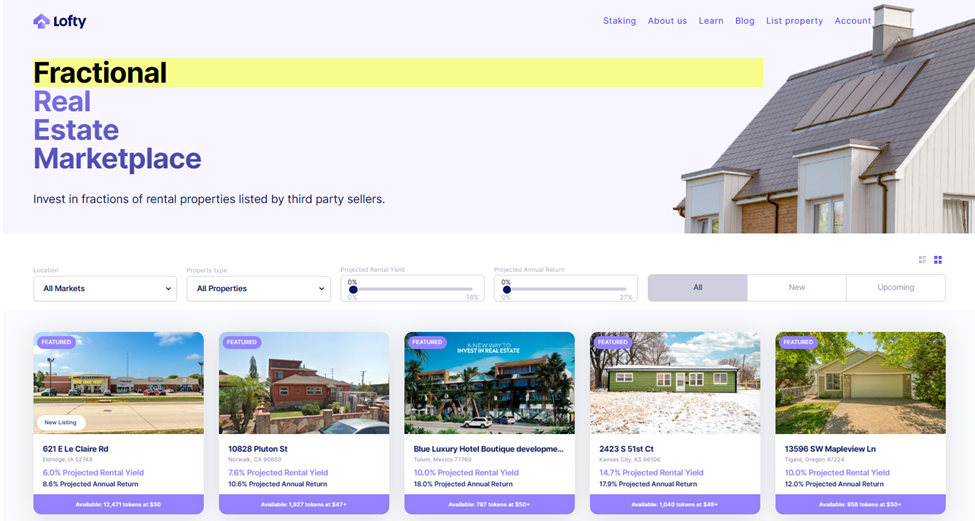
* **Thiếu thụt thông tin giữa người mua và bán**: Đối với các loại tài sản khác như cổ phiếu/trái phiếu, tìm kiếm thông tin giúp người mua đưa ra quyết định là tương đối dễ dàng. Việc này có thể thực hiện thông qua mạng hoặc các trang báo online như Bloomberg, The Morning Star. Tuy nhiên trong BĐS thì nguồn thông tin chính của người mua là do người bán cung cấp Þ Dẫn tới người bán có thể cung cấp thông tin sai lệch, thiếu thụt và làm ảnh hưởng tới quá trình thượng lượng giá
* **Thủ tục giấy tờ tốn thời gian và chi phí:** Quy trình giấy tờ phức tạp và liên quan tới nhiều bên thứ ba khiến cho giao dịch mất nhiều thời gian, tiền bạc. Việc xử lý nhiều giấy tờ cũng tăng nguy cơ xảy ra sai sót và làm trễ tiến độ giao dịch.
* **Chi phí tham gia vào thị trường BĐS ban đầu cho người mua/bán là rất cao**: Chi phí đầu vào lớn đối với người mua/bán là một thách thức lớn. Người mua phải đối mặt với khoản chi phí như tiền cọc, phí cho trung tâm môi giới. Trong khi đó người bán có thể phải chịu các chi phí liên quan đến việc chuẩn bị và quảng bá bất động sản



Þ Mỗi một bên thứ ba đều yêu cầu lượng phí nhất định, dẫn đến chi phí giao dịch cuối cùng cho cả người mua lẫn bán là rất cao và tốn kém thời gian do phụ thuộc vào bên thứ ba

## **Nội dung đề tài và giới thiệu nền tảng Lofty**

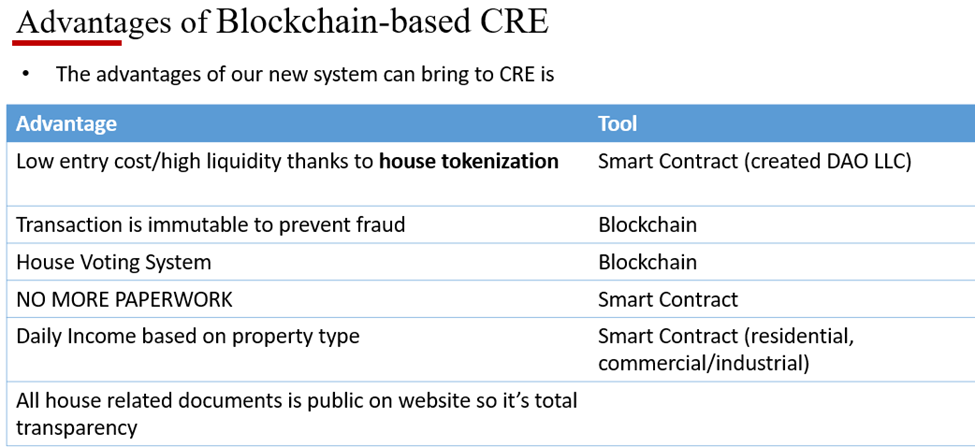
Thấy được những thách thức trong giao dịch BĐS truyền thống, nhóm của chúng tôi muốn quyết định triển khai hệ thống bất động sản được xây dưng trên nền tảng Blockchain Hyperledger Fabric



Hệ thống của nhóm **được xây dựng và lấy cảm hứng dựa trên nền tảng Lofty** [Fractional Real Estate Marketplace | Lofty](https://www.lofty.ai/). Lofty được ra đời vào năm 2020, sử dụng nền tảng blockchain của Algorand cho các giao dịch BĐS. Điều này giúp mang lại một số lợi thế vượt trội so với giao dịch BĐS truyền thống như

* Giảm thiểu số lượng bên thứ ba trong các giao dịch
* Cung cấp thông tin minh bạch và giảm thời gian
* Giảm chi phí đầu vào cho người mua/bán dựa và tăng tính thanh khoản dựa vào House Tokenization [What is tokenized real estate? A beginner's guide to digital real estate ownership (cointelegraph.com)](https://cointelegraph.com/learn/what-is-tokenized-real-estate)

Þ Giúp cho giao dịch BĐS hiệu quả và tiện lợi cho cả người mua và bán



# **CHƯƠNG 2 - PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

[VinhQuocTran/finalterm\_real\_estate\_blockchain (github.com)](https://github.com/VinhQuocTran/finalterm_real_estate_blockchain)

## **Đặc tả yêu cầu chức năng**

|  |  |
| --- | --- |
| **Yêu cầu chức năng** | Hệ thống hỗ trợ người dùng đăng ký/đăng nhập  Hệ thống hỗ trợ quản lý property  Hệ thống hỗ trợ đưa property lên marketplace  Hệ thống hỗ trợ mua và bán token  Hệ thống hỗ trợ tìm property theo nhu cầu  Hệ thống hỗ trợ xem token đang sở hữu và rút tiền thuê  Hệ thống hỗ trợ xem các giao dịch |

Bảng 1: Yêu cầu chức năng phía User

|  |  |
| --- | --- |
| **Yêu cầu chức năng** | Hệ thống hỗ trợ verify và listing property  Hệ thống hỗ trợ xem lịch sử verify và listing property |

Bảng 2: Yêu cầu chức năng phía Admin

## **Đặc tả yêu cầu phi chức năng**

|  |  |
| --- | --- |
| **Yêu cầu phi chức năng** | Hiệu suất hoạt động hệ thống  Sao lưu dữ liệu  Cho phép người dùng tương tác dễ dàng  Bảo mật thông tin khách hàng  Khả năng tương thích với đa nền tảng  Khả năng bảo trì hệ thống |

**Tính bảo mật và phân quyền cho user**: Hệ thống yêu cầu user phải đăng nhập bằng multifactor-authentication trước khi sử dụng. Giao diện hệ thống phụ thuộc vào tài khoản mà user đăng nhập.

**Hiệu năng cao (performance):** Với điều kiện kết nốt mạng ổn định, hệ thống phải có khả năng xử lý requests của ít nhất 100.000 users cùng 1 lúc đăng nhập và sử dụng

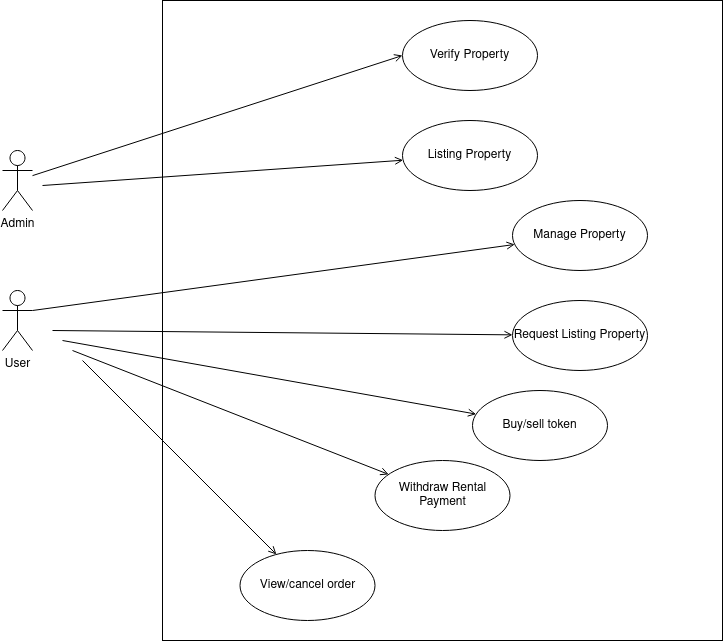
**Khả năng mở rộng (scalability):** Nếu quy mô công ty và số lượng user tăng lên, hệ thống phải có khả năng mở rộng tới tối đa 10.000.000 users cùng 1 lúc đăng nhập và sử dụng

**Khả năng tương thích với đa nền tảng (compatiablity):** Hệ thống cho phép đăng nhập trên nhiều nền tảng hệ điều hành khác nhau như Window, Android, IOS và nhiều thiết bị như PC, Tablet, Smartphone

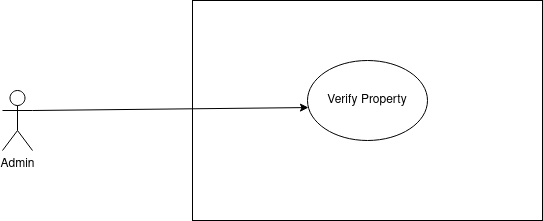
**Khả năng bảo trì hệ thống (maintainability):** Hệ thống phải được viết dưới dạng từng module, mỗi module là độc lập và không phụ thuộc vào nhau nhằm hỗ trợ mục đích phát triển và bảo trì dễ dàng

**Tính sẵn dùng (availability):** Hệ thống phải hoạt động 24/7 với tính khả dụng tối thiểu (minium availability) là 95%. Thời gian downtime hằng tuần không quá 5 tiếng

## **Lược đồ và đặc tả chức năng (2)**



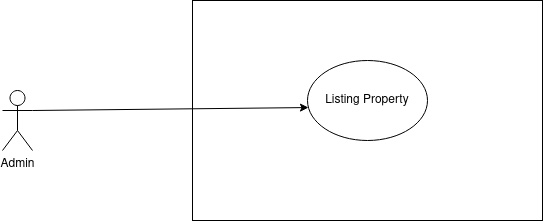
Hình: UseCase tổng quát



Hình 1: Lược đồ Usecase Admin verify property

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-1 | |
| **Use Case** | Verify Property | |
| **Scenario** | Chọn chức năng quản lý property | |
| **Trigger Event** | Admin chọn chức năng quản lý property | |
| **Description** | Admin đăng nhập vào hệ thống để chọn chức năng verify property | |
| **Actor(s)** | Admin | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện đăng nhập | |
| **Post-condition(s)** | Admin đăng nhập thành công vào hệ thống | |
| **Flow of event** | Actor | System |
| 1. Chọn mục Property tại sidebar | 1.1 Chuyển sang giao diện quản lý Property  1.2 Fetch danh sách các property được yêu cầu verify từ user. |
| 2. Chọn nút Verify  2.1 Chọn pass và nhấn accept  2.2 Chọn fail và nhấn accpet | 2. Hệ thống hiển thị modal gồm option: pass hoặc fail  2.1 Hệ thống cập nhật property đã được verified  2.2 Hệ thống cập nhật property chưa được verified và gửi mail về cho user |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

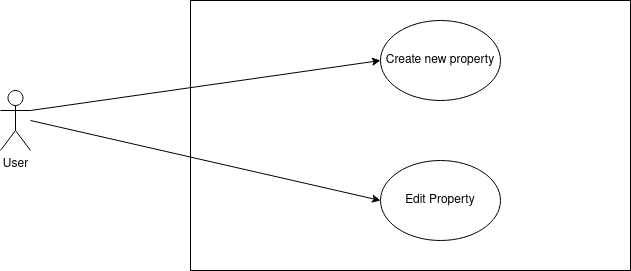
Bảng 3: Usecase Verify Property



Hình 2: Admin listing property

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-2 | |
| **Use Case** | Listing Property | |
| **Scenario** | Chọn chức năng quản lý property | |
| **Trigger Event** | Admin chọn chức năng quản lý property | |
| **Description** | Admin đăng nhập vào hệ thống để chọn chức năng listing property | |
| **Actor(s)** | Admin | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện đăng nhập | |
| **Post-condition(s)** | Admin đăng nhập thành công vào hệ thống | |
| **Flow of event** | Actor | System |
| 1. Chọn mục Property tại sidebar | 1.1 Chuyển sang giao diện quản lý Property  1.2 Fetch danh sách các property được yêu cầu listing từ user |
| 2. Chọn nút Listing  2.1 Chọn pass và nhấn accept  2.2 Chọn fail và nhấn accpet | 2. Hệ thống hiển thị modal gồm option: pass hoặc fail  2.1 Hệ thống cập nhật property đã được listed  2.2 Hệ thống cập nhật property chưa được listed và gửi mail về cho user |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

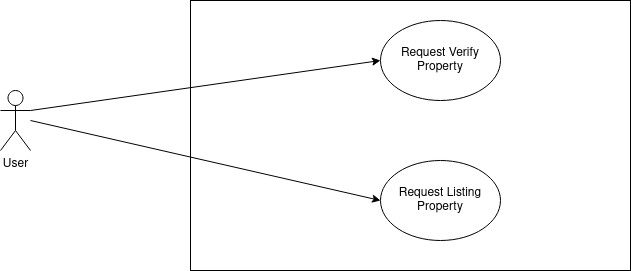
Bảng 4: Listing Property



Hình 3: User Manage Property

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-3 | |
| **Use Case** | Manage Property | |
| **Scenario** | Chọn tạo mới một property | |
| **Trigger Event** | User nhấn nút chọn tạo mới một property | |
| **Description** | User đăng nhập vào hệ thống để chọn chức năng tạo mới property | |
| **Actor(s)** | User | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện chức năng | |
| **Post-condition(s)** | User đăng nhập thành công vào hệ thống và đang ở trang My Property | |
| **Flow of event** | Actor | System |
| 1. Nhấn nút tạo mới property  1.2 Nhấn submit | 1.1 Hiển thị modal để nhập thông tin property  1.3 Hệ thống kiểm tra nếu thiếu thông tin sẽ báo lỗi. Ngược lại, sẽ tạo mới property và hiển thị lên giao diện |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

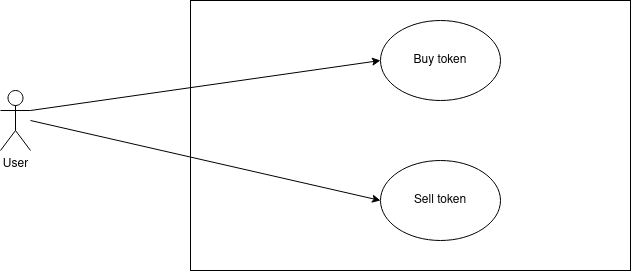
Bảng 5: User Manage Property



Hình 4: User Request Listing Property

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-4 | |
| **Use Case** | Listing Property | |
| **Scenario** | Yêu cầu đưa property lên marketplace | |
| **Trigger Event** | User nhấn vào nút listing property nếu nó khả dụng  User nhấn vào nút listing property | |
| **Description** | User nhấn vào nút verify property trên property card. Sau khi property được verify thì nút listing property sẽ khả dụng. Khi người dùng nhấn vào thì sẽ chọn tiếp các dịch vụ của bên thứ 3. Cuối cùng nhấn accept và đợi phản hồi từ admin. | |
| **Actor(s)** | User | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện chức năng | |
| **Post-condition(s)** | User đăng nhập thành công vào hệ thống và đang ở trang My Property. Đã có property trong danh sách | |
| **Flow of event** | Actor | System |
| 1. User nhấn nút verify tại property card  1.2 User nhấn accept | 1.1 Hiển thị modal để confirm yêu cầu từ user  1.3 Hệ thống chuyển trạng thái của property sang pending. |
| 2.1 User nhấn listing property  2.3 Nhấn chọn các dịch vụ và chọn accept | 2. Khi được verified bởi Admin thì hệ thống chuyển trạng thái property sang verified và nút listing property sẽ khả dụng.  2.2 Hệ thống hiện modal yêu cầu chọn các dịch vụ của bên thứ 3  2.4 Hệ thống kiểm tra và trừ cash balance của người dùng, chuyển trạng thái property sang pending. Khi các dịch vụ được mark passed bởi admin thì property sẽ lên marketplace. Nếu không thể listing thì sẽ thông báo về mail cho người dùng kèm lí do. |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

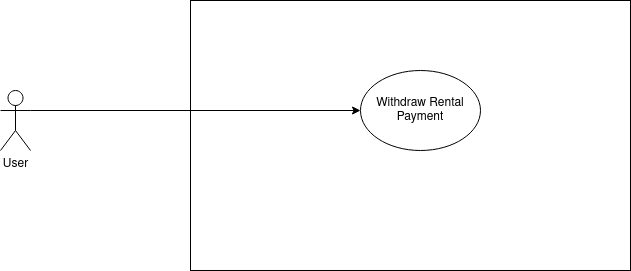
Bảng 6: Use case Listing Property



Hình 5: Usecase buy or sell token

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-5 | |
| **Use Case** | Sell/buy token | |
| **Scenario** | Bán token sau khi property được đưa lên marketplace | |
| **Trigger Event** | Tại giao diện chi tiết property, người dùng nhấn nút sell | |
| **Description** | Người dùng nhấn nút sell property, chọn số lượng token và giá | |
| **Actor(s)** | User | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện chức năng | |
| **Post-condition(s)** | User đăng nhập thành công vào hệ thống và đang ở trang chi tiết property. Current token phải khả dụng. | |
| **Flow of event** | Actor | System |
| 1. User nhấn nút sell/buy  1.2 User nhập đầy đủ thông tin và nhấn submit | 1.1 Hiển thị modal yêu cầu user nhập số lượng token và giá  1.3 Hệ thống cập nhật token balance của user và mục order book của property |
|  | 2. Nếu tồn tại một cặp order khớp lệnh thì sẽ thực hiện transaction và cập nhật trạng thái completed trong mục my order |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

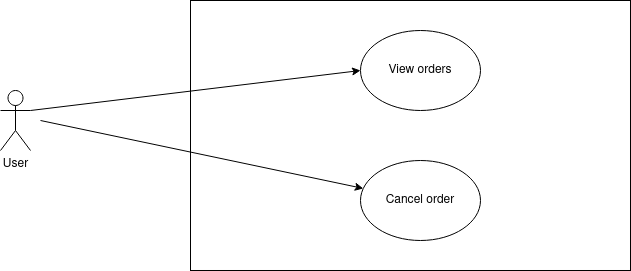
Bảng 7: Use case buy or sell token



Hình 6: Use case Withdraw Rental Payment

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-6 | |
| **Use Case** | Withdraw Rental Payment | |
| **Scenario** | Rút tiền thuê hàng từ các token đã bán | |
| **Trigger Event** | Tại giao diện token ownership and rental payment | |
| **Description** | Người dùng nhấn nút withdraw để chuyển lợi nhuận về cash balance | |
| **Actor(s)** | User | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện chức năng | |
| **Post-condition(s)** | User đăng nhập thành công vào hệ thống và đang ở trang token ownership and rental payment | |
| **Flow of event** | Actor | System |
| 1. User nhấn nút withdraw | 1.1 Hệ thống chuyển số dư về tài khoản user, cập nhật cash balance và số tiền thuê của các token.  1.3 Hệ thống hiển thị thông báo thành công |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

Bảng 8: Use case withraw rental payment



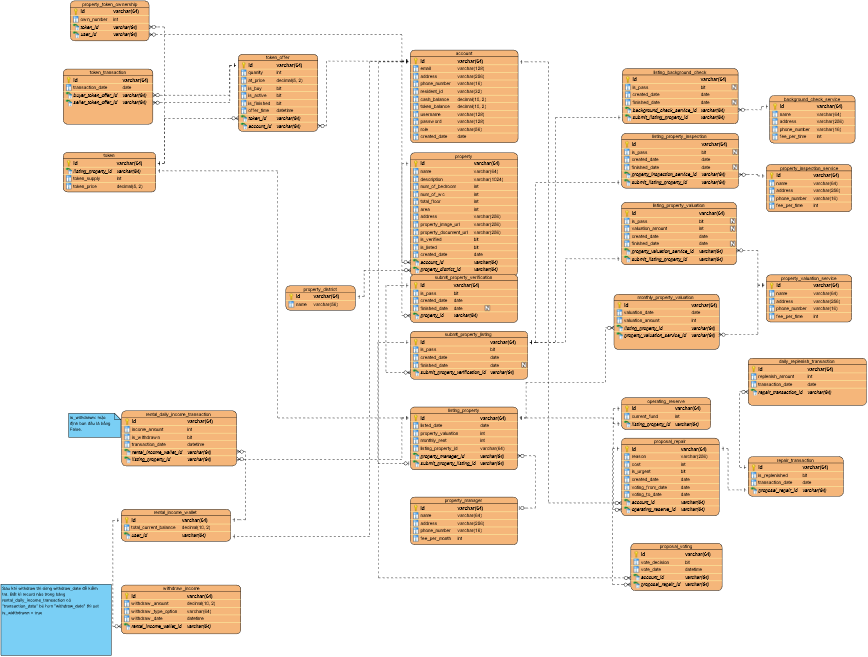
Hình 7: Use case view or cancel orders

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC-6 | |
| **Use Case** | View/cancel order | |
| **Scenario** | User có thể xem và hủy các order | |
| **Trigger Event** | Nhấn vào mục my order để xem danh sách các order. Nhấn vào nút cancel để hủy order | |
| **Description** | User nhấn vào mục my order để xem chi tiết các order. Sau đó, nếu muốn hủy order thì nhấn nút cancel | |
| **Actor(s)** | User | |
| **Pre-condition(s)** | Tài khoản khả dụng và được kết nối internet khi thực hiện đăng nhập | |
| **Post-condition(s)** | User đã đăng nhập thành công vào hệ thống | |
| **Flow of event** | Actor | System |
| 1. User nhấn vào mục my order | 1.1 Hiển thị danh sách các order |
| 2. User nhấn cancel order | 2.1 Hệ thống cập nhật order status |
| **Exception** | Hệ thống xảy ra lỗi trong quá trình xử lý dữ liệu và hiển thị thông báo lỗi | |

Bảng 9: Use case view and cancel order

## **Lược đồ và đặc tả quan hệ thực thể (ER diagram/specification)**

* Physical ERD của hệ thống



* Đối với đặc tả thì xem trong Github Repo theo đường dẫn: **document/erd\_specification**

# **CHƯƠNG 3 - HIỆN THỰC HỆ THỐNG**

## **Các công nghệ sử dụng trong đề tài**

* Git
* Docker & docker-compose
* NodeJS
* JQ >= 1.6
* Go version 20

Kiến trúc hệ thống:

Module Admin

* Hoạt động: Gửi request tới các APIs dành cho admin từ tầng Restful API
* Vai trò chính: Cho phép admin xác thực tài sản , Listing tài sản và cập nhật giá của tài sản.

Module Lotfy

* Hoạt động: Gửi request tới các APIs dành cho user từ tầng Restful API
* Vai trò chính: user có thể đăng bán tài sản, mua các tài sản đã được listing, kiểm tra các giao dịch đã được thực hiện, thay đổi thông tin, nhận tiền lời hàng tháng từ các tài sản đã mua.

Module Application (RESTful API)

* Hoạt động: Nhận request từ module admin hoặc lofty sau đó tiến hành chạy cái smart contract để lấy dữ liệu từ ledger hoặc chạy những query để lấy dữ liệu từ database, sau đó trả về response.
* Vai trò chính: Chứa toàn bộ APIs của hệ thống.

Module Chaincode

* Hoạt động: Chứa các logic giao dịch, các hàm để truy vấn dữ liệu từ ledger.
* Vai trò chính: Giúp tầng RESTful tương tác với Ledger để truy vấn dữ liệu.

## **Cách setup và chạy hệ thống**

**Bước 1**: Tải script này:

curl -sSLO <https://raw.githubusercontent.com/hyperledger/fabric/main/scripts/install-fabric.sh> && chmod +x install-fabric.sh

**Bước 2**: Tải hyperledger fabric version 2.5.5

./install-fabric.sh --fabric-version 2.5.5 binary

**Bước 3**: Clone project từ github

git clone <https://github.com/VinhQuocTran/finalterm_real_estate_blockchain.git>

**Bước 4**: Chạy mạng blockchain

* Vào thư mục website/src/test-network
* Khởi chạy mạng thử nghiệm Fabric bằng tập lệnh shell network.sh.

./network.sh up createChannel -c mychannel -ca

* Triển khai smart contract với javascript

./network.sh deployCC -ccn basic -ccp ../src/chaincode -ccl javascript

**Bước 5**: Chạy project Application

* Vào thư mục website/src/application

npm build   
npm run start

**Bước 6**: Chạy project Lotfy-clone

* Vào thư mục website/src/lotfy-clone

npm install --legacy-peer-deps

npm run start

**Bước 7**: Chạy project admin

* Vào thư mục website/src/admin

npm install --legacy-peer-deps

npm run start

**Bước 8**(Optional):

* Tạm dừng mạng blockchain

docker stop $(docker ps -a -q)

* Chạy lại mạng blockchain

docker restart $(docker ps -a -q)

**Bước 9**: Xóa tất cả container trong docker

* Vào thư mục website/src/test-network:

./network.sh down

* Muốn chạy lại mạng blockchain quay lại bước 4.

## **Giao diện và demo các chức năng chính**

* [Lofty Application Demo (youtube.com)](https://www.youtube.com/watch?v=hpG0Eyt1if8)

A screenshot of a computer

Description automatically generated

# **CHƯƠNG 4 - KẾT LUẬN**

## **Hướng phát triển**

* Hệ thống phía trên chỉ là phiên bản demo/POC (Proof of concept) nhằm chứng minh là dự án hoàn toàn có tính khả thi. Vậy nên để phát triển lên thì ta có thể thêm và hoàn chỉnh các chức năng như: Giao diện của User và Admin, Cho phép xem lịch sử giá nhà và dự đoán lợi nhuận theo từng năm, phát triển voting/governance system cho từng căn nhà, thêm trang hỗ trợ User viết ticket và customer support, tích hợp hệ thống nạp và rút tiền từ coin wallet/visa/paypal/mastercard.
* Ta có thể xây dựng Data Warehouse bằng các dịch vụ cloud như Azure SQL Data Warehouse, AWS Redshift. Data warehouse này sẽ lấy transactional và master data của hệ thống phía trên để tổng hợp. Cuối cùng là kết nối với các BI tools như PowerBI, Tableau để vẽ dashboard nhằm hỗ trợ nhóm trong việc đưa ra các quyết định kinh doanh
* Ngoài ra có thể dùng Data Warehouse phía trên để phát triển các mô hình học máy để dự đoán giá/lợi nhuận của từng căn nhà dựa vào thông tin người dùng cho trước như số tầng, số phòng ngủ/phòng tắm,..

## **Ý nghĩa thực tiễn**

* Thông qua dự án lần này, nhóm chúng tôi đã giúp cho từng thành viên trong nhóm hiểu được thêm về cách thức hoạt động của nền tảng giao dịch BĐS, Data model trong đó thường bao gồm những thông tin gì và liên kết với nhau như thế nào, quy trình mua/bán nhà như thế nào.
* Các thành viên còn biết được thêm về quy trình phát triển phần mềm nói chung và nền tảng Hyperledger Fabric nói riêng, các kiến trúc cũng như phương pháp để phát triển hệ thống dựa trên công nghệ Blockchain

**BẢNG PHÂN CHIA CÔNG VIỆC**

|  |  |  |  |
| --- | --- | --- | --- |
| **Số thứ tự** | **Full name** | **Student ID** | **Role** |
| 1 | Trần Quốc Vinh | 52000823 | Data architect/consultant |
| 2 | Trần Văn Thảo | 52000805 | Fullstack developer |
| 3 | Trần Minh Trí | 52000815 | Fullstack developer |

**CHƯƠNG 1 – TỔNG QUAN ĐỀ TÀI**

* Các thách thức trong lĩnh vực buôn bán bất động sản (bđs) truyền thống (1)
* Nội dung đề tài và giới thiệu nền tảng Lofty (1)

**CHƯƠNG 2 – PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

* Đặc tả yêu cầu chức năng (2)
* Đặc tả yêu cầu phi chức năng (1)
* Lược đồ và đặc tả chức năng (Use-case diagram/specification) (2)
* Lược đồ và đặc tả quan hệ thực thể (ER diagram/specification) (1)

**CHƯƠNG 3 – HIỆN THỰC HỆ THỐNG**

* Các công nghệ sử dụng trong đề tài (3)
* Cách setup và chạy hệ thống (3)
* Giao diện và demo các chức năng chính (2: làm video)

**CHƯƠNG 4 – KẾT LUẬN**

* Hướng phát triển (1)
* Ý nghĩa thực tiễn (1)

**TÀI LIỆU THAM KHẢO**

[Fractional Real Estate Marketplace | Lofty](https://www.lofty.ai/)