BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN

Đề tài

XÂY DỰNG HỆ THỐNG MUA BÁN THÔNG TIN

Sinh viên: Nguyễn Tấn Đạt

Mã số sinh viên: C1800005

Khóa: K44

Cần Thơ, 06/2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN

Đề tài

XÂY DỰNG HỆ THỐNG MUA BÁN THÔNG TIN

Người hướng dẫn
TS. Trần Công Án

Sinh viên thực hiện
Nguyễn Tấn Đạt
Mã số: C1800005
Khóa: K44

Cần Thơ, 06/2021

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

……
……
……

LỜI CẢM ƠN

Sự thành công nào mà không gắn liền với những sự hỗ trợ, giúp đỡ dù ít hay nhiều, dù trực tiếp hay gián tiếp của người khác. Trong suốt thời gian từ khi bắt đầu học tập ở ***Trường Đại Học Cần Thơ*** đến nay, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè.

Cho phép em xin gửi lời cảm ơn chân thành đến quý Thầy Cô ***Khoa Công Nghệ Thông Tin Và Truyền Thông - Trường Đại Học Cần Thơ***. Trong suốt thời gian qua, quý Thầy Cô đã tận tình chỉ dạy, truyền đạt những kiến thức cũng như kinh nghiệm quý báu cho em. Trong học kỳ này, Khoa đã tạo điều kiện cho em được làm đề tài luận văn, đó được xem như thành quả mà em có được khi học tập tại ngôi trường thân yêu này. Đồng thời em xin được bày tỏ lòng biết ơn sâu sắc của thầy ***Trần Công Án***, thầy là người đã hướng dẫn em hoàn thành đề tài của mình.

Do thời gian có hạn cũng như kinh nghiệm, kiến thức còn thiếu nhiều nên đề tài báo cáo này sẽ không tránh khỏi những thiếu sót, hạn chế nhất định. Em rất mong nhận được nhận xét góp ý quý báu của Thầy Cô để kiến thức của em trong lĩnh vực này hoàn thiện tốt hơn.

Cảm ơn anh Phạm Trường An đã giúp đở tận tình trong việc giải đáp những câu hỏi về vấn đề kỹ thuật cũng như hỗ trợ về ý tưởng đề tài.

Cuối lời, em xin kính chúc quý Thầy Cô ***Trường Đại Học Cần Thơ*** luôn luôn mạnh khỏe để tiếp tục truyền đạt lại những kiến thức và kinh nghiệm cho thế hệ mai sau.

Em xin chân thành cảm ơn !

Cần Thơ, ngày 07 tháng 06 năm 2021

Sinh viên thực hiện

Nguyễn Tấn Đạt

MỤC LỤC

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN i
LỜI CẢM ƠN ii
MỤC LỤC iii
DANH MỤC HÌNH ẢNH vii
DANH MỤC BẢNG ix
TÓM TẮT x
ABSTRACT xi
PHẦN GIỚI THIỆU 1
1. Đặt vấn đề 1
2. Hiện trạng 1
3. Mục tiêu đề tài 1
4. Đối tượng và phạm vi nghiên cứu 2
4.1. Đối tượng người sử dụng nhắm đến 2
4.2. Phạm vi nghiên cứu 2
5. Nội dung nghiên cứu 2
5.1. Về lý thuyết 2
5.2. Về lập trình 2
5.3. Về người dùng và chức năng 2
6. Bố cục đề tài 3
PHẦN NỘI DUNG 5
CHƯƠNG 1 - CƠ SỞ LÝ THUYẾT 5
1. NextJS và Server Side Rendering 5
2. NextJS Web API 5
3. ReactJS 5
4. React Native 6
5. PostgreSQL 6
6. Firebase Authentication 6
6.1. Giới thiệu Firebase Authentication 6
6.2. Phương thức đăng nhập bằng Google 7
6.3. Control Access with Custom Claims và chức năng xác thực bằng token 7
7. Prisma 7
8. Phương thức nạp tiền tự động bằng MoMo 8

8.1. Giới thiệu MoMo 8
8.2. MoMo API 8
8.3. Mã giao dịch MoMo 8
8.4. Cổng API https://momofree.apimienphi.com 8
8.5. Kết hợp API vào nạp tiền tự động 10
9. Các công nghệ khác có sử dụng 10
9.1. Tailwinds CSS 10
9.2. Docker 10
CHƯƠNG 2 - THIẾT KẾ GIẢI PHÁP 11
2.1. Phân tích thiết kế mô hình 11
2.1.1. Mô tả hệ thống 11
2.1.2. Thiết kế mô hình 11
2.1.3. Sơ đồ chức năng người quản trị 13
2.1.4. Sơ đồ chức năng người dùng 14
2.2. Phân tích và thiết kế cơ sở dữ liệu 14
2.2.1. Xây dựng các tập thực thể 14
2.2.1.1. Thực thể Project 15
2.2.1.2. Thực thể User 15
2.2.1.3. Thực thể UserPaymentMethod 16
2.2.1.4. Thực thể UserDepositRecorder 16
2.2.1.5. Thực thể UserWithdrawRecorder 17
2.2.1.6. Thực thể MomoTransaction 17
2.2.1.7. Thực thể AvailableWithdrawMethod 18
2.2.1.8. Thực thể AvailableDepositMethod 18
2.2.1.9. Thực thể ProjectTransactionRecorder 18
2.2.1.10. Thực thể Media 19
2.2.1.11. Thực thể MediaMap 19
2.2.2. Mô tả các mối quan hệ trọng tâm 20
2.2.2.1. Mối quan hệ khi người dùng gửi ảnh cho dự án 20
2.2.2.2. Mối quan hệ khi người dùng nạp tiền 21
2.2.2.3. Mối quan hệ thêm phương thức thanh toán 21
2.2.2.4. Mối quan hệ khi dùng rút tiền 21
2.2.3. Xây dựng mô hình PDM (Physical Data Model) 22
2.2.4. Xây dựng mô hình LDM (Logical Data Model) 23

CHƯƠNG 3 - GIAO DIỆN HỆ THỐNG VÀ CHỨC NĂNG 24
3.1. Giao diện cho người dùng web 24
3.1.1. Header 24
3.1.2. Trang chủ 24
3.1.3. Trang đăng nhập 25
3.1.4. Trang Profile 26
3.1.5. Trang tạo mới dự án 28
3.1.6. Trang xem chi tiết một dự án và gửi hình ảnh 29
3.1.7. Trang dự án của tôi 31
3.1.8. Trang danh sách ảnh đã nhận của dự án 32
3.1.9. Trang thanh toán hình ảnh 34
3.1.10. Trang duyệt hình ảnh trên bản đồ 35
3.1.11. Trang đóng góp của tôi 36
3.2. Giao diện cho người quản trị 36
3.2.1. Trang quản lý dự án 36
3.2.2. Trang quản lý ảnh đóng góp 37
3.2.3. Trang quản lý thành viên 37
3.2.4. Trang quản lý nạp và rút tiền 38
3.3. Giao diện cho người dùng di động 38
CHƯƠNG 4 - ĐÁNH GIÁ VÀ KIỂM THỬ 40
4.1. Mục tiêu 40
4.2. Phạm vi kiểm thử 40
4.3. Quy trình kiểm thử 40
4.4. Môi trường kiểm thử 40
4.4.1. Môi trường chạy ứng dụng 40
4.4.2. Chi tiết cấu hình máy chạy ứng dụng 40
4.5. Các chức năng kiểm thử 41
4.6. Một số chức năng không được kiểm thử 41
4.7. Tiêu chí kiểm thử 41
4.8. Kiểm thử chức năng 41
4.8.1. Đăng nhập 41
4.8.2. Nạp tiền vào tài khoản 42
4.8.3. Tạo lệnh rút tiền 43
4.8.4. Tạo dự án mới 44

4.8.5. Gửi ảnh cho dự án 44
4.8.6. Gửi ảnh trên bản đồ trên thiết bị di động 45
4.8.7. Thanh toán ảnh đã duyệt 46
PHẦN KẾT LUẬN 47
1. Kết luận 47
1.1. Kết quả đạt được 47
1.2. Ưu điểm 47
1.3. Hạn chế 47
2. Hướng phát triển 47
TÀI LIỆU THAM KHẢO 49

DANH MỤC HÌNH ẢNH

Hình 1: Chi tiết Response....9
Hình 2: Sơ đồ usecase người dùng chưa đăng nhập....11
Hình 3: Sơ đồ usecase thành viên sử dụng di động....12
Hình 4: Sơ đồ usecase Admin....12
Hình 5: Sơ đồ usecase thành viên sử dụng web....13
Hình 6: Sơ đồ chức năng người quản trị....13
Hình 7: Sơ đồ chức năng người dùng web....14
Hình 8: Sơ đồ người dùng di động....14
Hình 9: Mối quan hệ gửi ảnh cho dự án....20
Hình 10: Mối quan hệ khi người dùng nạp tiền....21
Hình 11: Mối quan hệ thêm phương thức thanh toán....21
Hình 12: Mối quan hệ khi dùng rút tiền....21
Hình 13: Mô hình PDM....22
Hình 14: Mô hình LDM....23
Hình 15: Header người dùng chưa đăng nhập....24
Hình 16: Header người dùng đã đăng nhập....24
Hình 17: Header người quản trị....24
Hình 18: Giao diện trang chủ....24
Hình 19: Trang đăng nhập....25
Hình 20: Pop-up đăng nhập....25
Hình 21: Trang cá nhân....26
Hình 22: Chỉnh sửa trang cá nhân....26
Hình 23: Hướng dẫn nạp tiền....27
Hình 24: Hướng dẫn rút tiền....27
Hình 25: Lịch sử nạp/rút....28
Hình 26: Thêm phương thức thanh toán....28
Hình 27: Trang tạo mới dự án....29
Hình 28: Trang chi tiết dự án....30
Hình 29: Modal chọn ảnh....31

Hình 30: Trang dự án của tôi....31
Hình 31: Xem dự án đã tạo....32
Hình 32: Chỉnh sửa dự án....32
Hình 33: Xét duyệt hình ảnh....33
Hình 34: Hình ảnh đã duyệt....33
Hình 35: Thanh toán hình ảnh....34
Hình 36: Thông tin thanh toán hàng loạt....34
Hình 37: Thông tin thanh toán từng thành viên....35
Hình 38: Hiển thị thông tin hình ảnh trên bản đồ....35
Hình 39: Trang đóng góp của tôi....36
Hình 40: Trang quản lý dự án....37
Hình 41: Duyệt dự án....37
Hình 42: Trang quản lý hình ảnh đóng góp....37
Hình 43: Trang quản lý thành viên....37
Hình 44: Trang quản lý nạp và rút tiền....38
Hình 45: Hiển thị các lệnh nạp và thông tin người nạp....38
Hình 46: Giao diện trang chủ, trang cá nhân và trang chọn hình ảnh....39
Hình 47: Giao diện chụp ảnh trên bản đồ....39
Hình 48: Minh họa giao dịch không tồn tại....43
Hình 49: Minh họa mã giao dịch đã được sử dụng....43
Hình 50: Minh họa lỗi thiếu thông tin....44
Hình 51: Minh họa lỗi sai định dạng....45

DANH MỤC BẢNG

Bảng 1: Thực thể Project 15
Bảng 2: Thực thể User 15
Bảng 3: Thực thể UserPaymentMethod 16
Bảng 4: Thực thể UserDepositRecorder 17
Bảng 5: Thực thể UserWithdrawRecorder 17
Bảng 6: Thực thể MomoTransaction 17
Bảng 7: Thực thể AvailableWithdrawMethod 18
Bảng 8: Thực thể AvailableDepositMethod 18
Bảng 9: Thực thể ProjectTransactionRecorder 18
Bảng 10: Thực thể Media 19
Bảng 11: Thực thể MediaMap 19
Bảng 12: Kiểm thử chức năng đăng nhập 41
Bảng 13: Kiểm thử nạp tiền vào tài khoản 42
Bảng 14: Kiểm thử chức năng rút tiền 43
Bảng 15: Kiểm thử tạo dự án mới 44
Bảng 16: Kiểm thử gửi ảnh cho dự án 45
Bảng 17: Kiểm thử chức năng gửi hình trên di động 46
Bảng 18: Kiểm thử chức năng thanh toán 46

TÓM TẮT

Cùng với sự phát triển của công nghệ, dữ liệu số ngày càng chứng minh được tầm quan trọng trong việc tối ưu và phát triển sản phẩm một cách hoàn thiện nhất. Cũng vì đó mà nhu cầu để có được nguồn dữ liệu đúng với những yêu cầu đặc thù càng tăng cao.

Dự án Chợ Thông Tin ra đời như là một giải pháp để các kết nối những dự án cần nguồn tài nguyên với những người sẵn sàng tìm kiếm, sản xuất thông tin với một mức giá hợp lý. Với mục tiêu quan trọng là tạo được môi trường thuận lợi, minh bạch để cả người mua và người bán tin tưởng cung cấp tài nguyên của mình lên hệ thống. Thông qua website Chợ Thông Tin, người dùng có đầy đủ các công cụ để thực hiện quy trình mua bán như: Sử dụng tài khoản Google để đăng nhập mà không cần đăng ký, Tạo mới dự án theo các loại hình khác nhau, hệ thống đăng tải và xét duyệt hình ảnh trực quan, giao diện quản lý tối đa những tài nguyên của bản thân, cơ chế watermark ảnh để bảo vệ người bán, quy trình xét duyệt ảnh minh bạch cho cả 2 bên, chức năng nạp tiền tự động qua MoMo và cùng nhiều tính năng khác. Bên cạnh đó, người quản trị có giao diện riêng với toàn quyền quản lý giúp cho website có thể hoạt động một cách hiệu quả và an toàn. Ứng dụng còn hỗ trợ trên tảng Android, giúp tăng khả tiếp cận đến người dùng.

Các công nghệ được sử dụng trong dự án Chợ Thông Tin đã được minh chứng về sức mạnh bởi cộng đồng lập trình trên thế giới. Cụ thể, về phần giao diện, hệ thống sử dụng ReactJS để thực hiện các chức năng xử lý logic, NextJS để xử lý phần dynamic routing và Server Side Rendering, cuối cùng là Tailwinds CSS để làm giao diện hiển thị. Về phần xử lý đăng nhập, Firebase Authentication giúp hỗ trợ tối đa việc đăng nhập bằng tài khoản Google. Về phần xử lý back-end, hệ thống sử dụng Web API chạy trên nền NextJS và Prisma cho việc giao tiếp với cơ sở dữ liệu sử dụng Postgres. Cùng một số công nghệ phụ trợ khác.

Đề tài Ứng Dụng Chợ Thông Tin đã hoàn thành tương đối tốt các yêu cầu mà ban đầu đã đề ra. Hệ thống đáp ứng đầy đủ các chức năng và hoạt động tốt.

ABSTRACT

Along with the development of technology, digital data increasingly proves the importance of optimizing and developing products in the most complete way. Because of that, the need to get the right data source with specific requirements is also growing.

The Information Market Project was born as a solution to connect projects that need resources with people who are willing to find and produce information at a reasonable price. With the important goal of creating a favorable and transparent environment for both buyers and sellers to trust their resources to the system. Through the Information Market website, users have a full range of tools to carry out the buying and selling process such as: using a Google account to log in without registration, Create new projects of different types, an intuitive image uploading and review system, an interface to maximize your own resources, photo watermark mechanism for seller protection, transparent photo review process for both parties, automatic top-up via MoMo and more. In addition, the administrator has a separate interface with full management rights to help the website operate effectively and safely. The application also supports the Android platform, helping to increase accessibility to users.

The technologies used in the Information Market project have been proven in strength by the programming community around the world. Specifically, for the interface, the system uses ReactJS to perform logic processing functions, NextJS to handle dynamic routing and Server Side Rendering, and finally, Tailwinds CSS to make the display interface. As for login handling, Firebase Authentication helps to maximize login with Google accounts. As for the back-end processing, the system uses Web API running on NextJS and Prisma for communicating with the database using Postgres. And some other supporting technologies.

The Application of Information Market has completed relatively well the requirements that were initially set out. The system is fully functional and works well.

CAM KẾT KẾT QUẢ

Tôi xin cam kết luận văn này được hoàn thành dựa trên kết quả nghiên cứu của tôi và các kết quả này của nghiên cứu chưa được dùng cho bất cứ luận văn cùng cấp nào khác.

Tôi xin cam đoan mọi sự giúp đỡ cho việc thực hiện luận văn này đã được cảm ơn và các tài liệu tham khảo trong luận văn đã được ghi rõ nguồn gốc.

Cần Thơ, ngày 07 tháng 06 năm 2021

Sinh viên thực hiện

Nguyễn Tấn Đạt

PHẦN GIỚI THIỆU

1. Đặt vấn đề

Với xu thế phát triển của Internet hiện nay, dữ liệu đã trở thành 1 phần quan trọng trong định hướng chiến lược và nghiên cứu sản phẩm của bất kỳ một ngành thương mại nào. Và cũng vì thế, các công ty công nghệ lớn, các chính phủ hoặc thậm chí những startup nhỏ đều tham gia cuộc chiến tranh giành dữ liệu từ người sử dụng Internet. Các thông tin cá nhân, hình ảnh, hành vi của hàng tỉ người dùng trên thế giới đều được lưu trữ lại và biến thành một loại tài sản vô cùng giá trị. Nhưng điều đáng buồn là không phải công ty hoặc dự án nào cũng có cơ hội tiếp cận được nguồn dữ liệu này. Họ sẵn sàng trả tiền để có những tư liệu và hình ảnh phù hợp, nhưng mua từ các công ty khác là một điều khá nhạy cảm vì có liên quan đến chính sách riêng tư và bản quyền dữ liệu.

Ở phía ngược lại, những người sử dụng Internet là những người trực tiếp hàng ngày tạo ra kho dữ liệu đó. Họ không hoặc ít có nhận thức rằng các dữ liệu quý giá đó đang được cho đi một cách miễn phí và tự nguyện.

Từ đó ta có bài toán làm sao để kết nối được những người cần mua với những người đang nắm giữ dữ liệu nhưng lại chưa khai thác hết được tiềm năng. Đồng thời, cần phải có một hệ thống trung gian để đảm bảo quyền lợi đôi bên cũng như giải quyết các vấn đề đặc thù mà 1 sản phẩm dữ liệu số có như: quyền khai thác dữ liệu, thanh toán trung gian, đánh giá độ phù hợp dữ liệu, giải quyết tranh chấp,…

2. Hiện trạng

Hiện nay trên thế giới có khá nhiều các dự án giải quyết bài toán tương tự, có thể kể tên như Trove của Microsoft nhưng chỉ hỗ trợ sử dụng ở Mỹ. Còn tại việt nam hầu hết đều là những chợ đen, những người bán hình ảnh sưu tầm được trên Internet với độ an toàn cũng như chất lượng không được đảm bảo.

3. Mục tiêu đề tài

Tạo ra được hệ thống có đủ các tính năng cơ bản về quản trị người dùng, bài viết, dữ liệu cũng như vấn đề chính là tạo điều kiện để bên bán và bên mua có môi trường mua bán minh bạch. Ngoài ra, hệ thống phải thiết kế dễ sử dụng, hỗ trợ đa nền tảng (Web, di động), giao diện người quản trị có toàn quyền quyết định trên hệ thống (Admin Portal) để kiểm duyệt các bài viết vi phạm tiêu chuẩn cộng đồng. Cung cấp thêm các chức năng hỗ trợ như: nạp tiền tự động qua momo, tự động thanh toán dữ liệu, preview dữ liệu để lựa chọn chấp nhận hoặc không, chụp hình kèm vị trí trên bản đồ. Hiện tại hệ thống chỉ tập trung vào loại dữ liệu là hình ảnh, tương lai sẽ mở rộng ra nhiều loại dữ liệu khác.

4. Đối tượng và phạm vi nghiên cứu

4.1. Đối tượng người sử dụng nhắm đến

Về phía người mua, đối tượng mà hệ thống nhắm đến là những dự án trí tuệ nhân tạo, học máy cần nguồn hình ảnh thích hợp để train. Mở rộng hơn là những người cần những hình ảnh về một chủ đề cụ thể nào đó, các hình ảnh của một số địa điểm trên bản đồ. Hoặc bất ai có nhu cầu mua ảnh.

Về phía người bán ảnh, đối tượng tiềm năng là những người có thời gian rảnh muốn kiếm thêm thu nhập. Họ có thời gian đi săn tìm ảnh theo đúng nhu cầu dự án. Hoặc đơn giản là những người có kho ảnh lớn muốn khai thác thêm giá trị.

4.2. Phạm vi nghiên cứu

Phạm vi nghiên cứu là bất kỳ ai có kết nối Internet và có nhu cầu bán hoặc mua hình ảnh hợp pháp, trên hai nền tảng di động và Web.

5. Nội dung nghiên cứu

Nghiên cứu về quy trình làm sao để đưa ảnh từ người bán đến người mua 1 cách dễ dàng và chi tiết nhất. Đảm bảo tiêu chí an toàn của hình ảnh người bán cũng như những thông tin mà server gửi về cho client.

5.1. Về lý thuyết

- Tìm hiểu khả năng và hạn chế của NextJS và cách sử dụng các tính năng của NextJS như: Dynamic Routes, Server Side Rendering, API Routes,...

- Tìm hiểu Firebase Authentication và xử lý xác thực bằng tài khoản Google.

- Nghiên cứu ReactJS, Javascript, Tailwinds CSS, HTML vào xây dựng giao diện website.

- Nghiên cứu React Native để xây dựng ứng dụng di động.

- Nghiên cứu sử dụng PostgreSQL và Prisma cho cơ sở dữ liệu.

5.2. Về lập trình

- Xây dựng giao diện, xử lý các logic, tuần tự, truy vấn cơ sở dữ liệu, tiền xử lý dữ liệu, hiển thị theo giao diện đã xây dựng.

- Xây dựng các API truy vấn dữ liệu thô và trả về dạng Json theo từng request của chức năng. Thực hiện các lệnh cập nhật, thêm, sửa, xóa dữ liệu theo yêu cầu.

- Xây dựng Admin Portal, quản lý tất cả các nội dung như: bài viết, hình ảnh, thành viên, thanh toán,... Admin có toàn quyền duyệt bài, thêm, sửa, xóa mọi nội dung trên trang web.

5.3. Về người dùng và chức năng

- Người dùng không có tài khoản:

+ Đăng nhập bằng tài khoản Google.

+ Xem yêu cầu của các dự án đã được duyệt.

- Thành viên:

+ Đăng nhập/đăng xuất.

+ Nạp tiền tự động qua MoMo.

+ Tạo lệnh rút tiền.

+ Chỉnh sửa thông tin trang cá nhân.

+ Tạo bài viết mới.

+ Bán ảnh cho các dự án đã duyệt.

+ Xem các ảnh đã gửi xét duyệt.

+ Xét duyệt, thanh toán các ảnh mà người khác đã gửi cho dự án đã tạo.

+ Xem trạng thái, xóa, sửa, tạm dừng các dự án đã tạo.

- Admin

+ Tất cả các quyền của thành viên.

+ Quản lý tất cả các nội dung có trên web.

+ Chỉnh sửa bất kỳ thông tin của dự án, thông tin cá nhân.

+ Tạm dừng hoạt động của thành viên.

+ Xem chi tiết, xử lý các lệnh thanh toán.

6. Bố cục đề tài

Nội dung quyển luận văn gồm các phần sau:

- Phần mở đầu: Lời cảm ơn, mục lục, danh mục biểu bảng, danh mục hình, các ký hiệu và từ viết tắt, tóm tắt, cam kết kết quả

- Phần giới thiệu: Nêu lên vấn đề cần giải quyết và phạm vi của đề tài. Qua đó lên kế hoạch và phương pháp thực hiện. Phần giới thiệu bao gồm những nội dung chính: Đặt vấn đề, hiện trạng, mục tiêu đề tài, đối tượng và phạm vi nghiên cứu, nội dung nghiên cứu.

- Phần nội dung: Gồm bốn chương, trình bày các nội dung chính của luận văn một cách chi tiết, cách giải quyết và kết quả đạt được.

+ Chương 1: Cơ sở lý thuyết về các công nghệ liên quan .

+ Chương 2: Thiết kế giải pháp và mô tả tổng quan hệ thống, thiết kế kiến trúc tổng thể, cơ sở dữ liệu.

+ Chương 3: Trình bài giao diện chức năng, tác dụng và giải thích của các thành phần trên giao diện.

+ Chương 4: Kiểm thử và đánh giá gồm có mô tả mục tiêu, kế hoạch, các trường hợp kiểm thử và kết quả kiểm thử, từ đó đưa ra đánh giá đối với các chức năng của hệ thống.

- Phần kết luận: Trình bày kết quả đạt được của đề tài cũng như những ưu điểm và hạn chế mà đề tài chưa thực hiện được, ngoài ra đưa ra hướng phát triển sau này.

PHẦN NỘI DUNG

CHƯƠNG 1 - CƠ SỞ LÝ THUYẾT

1. NextJS và Server Side Rendering

Ở thời kỳ đầu, khi web mới phát triển, khái niệm Server Side Render (SSR) đã được biết tới. Khi một request được gửi tới trang web, server sẽ xử lý nội dung thành HTML, return lại cho browser và hiển thị lại lên màn hình. Nhưng dần dần cách thức hoạt động của SSR tỏ ra nặng nề hơn và bộc lộ những nhược điểm nhất định, đó cũng là lúc Client Side Render (CSR) ra đời để khắc phục những nhược điểm đó, điển hình là ReactJS. Tuy nhiên, CSR không thể thay thế hoàn toàn được SSR, mỗi cái đều có những điểm mạnh và yếu khác nhau, và chúng đều quan trọng.

NextJS ra đời với sứ mệnh kết hợp ReactJS, một framework vô cùng mạnh mẽ xử lý CSR thành một ứng dụng SSR. Với NextJS, bạn có thể viết mã ReactJS, chạy nó trên máy chủ (mạnh hơn thiết bị client thông thường - như điện thoại di động), rồi gửi HTML document đến client. Bên cạnh đó, NextJS còn có rất nhiều tính năng nổi trội như:

+ Tối ưu cho việc SEO website.

+ Pre-rendering các trang có trong route với Link.

+ Hệ thống hỗ trợ định tuyến trực quan, dễ sử dụng.

+ Kho thư viện khổng lồ do sử dụng ReactJS làm giao diện.

+ API Routes build-in.

2. NextJS Web API

Web API là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại. Dữ liệu được Web API trả lại thường ở dạng JSON hoặc XML thông qua các giao thức HTTP hoặc HTTPS.

Web API cho phép lấy nội dung từ bất kỳ website hoặc ứng dụng nào nếu được cho phép, tăng khả năng tương thích giữa các nền tảng khi chỉ sử dụng chung một Web API.

NextJS hỗ trợ viết API ngay trong ứng dụng mà không cần chạy server riêng.

3. ReactJS

ReactJS là một thư viện dùng ngôn ngữ Javascript sử dụng để xây dựng UI (User Interface) cho web theo đúng xu hướng Single Page Applications. ReactJS cho phép nhúng code HTML vào code Javascript nhờ vào JSX. Các thành phần chính của ReactJS là những Components UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. Cho phép những nhà phát triển web tạo ra giao diện người dùng nhanh chóng.

Tính năng mà ReactJS mang lại là:

- ReactJS cực kỳ hiệu quả: nó tạo ra cho chính nó DOM ảo - nơi mà các components thực sự tồn tại trên đó. Việc tạo ra DOM như vậy giúp cải thiện hiệu suất làm việc rất nhiều, khi có tính toán cần thay đổi hoặc cần cập nhật những gì lên DOM thì ReactJS đều tính toán trước và việc còn lại chỉ là thực hiện chúng lên DOM. Làm như vậy sẽ giúp cho ReactJS thực hiện những thao tác cần có trên một DOM mà không cần tốn thêm bất cứ chi phí nào.

- Việc sử dụng JSX giúp các đoạn code JS kết hợp dễ dàng hơn với HTML. Ngoài ra ta còn có thể đem đoạn code thêm vào trong hàm render mà không cần phải thực hiện việc nối chuỗi.

- Kho thư viện đồ sộ cùng với cộng đồng sử dụng đông đảo. giúp tìm giải pháp cho các vấn đề một cách nhanh chóng.

4. React Native

React Native là một framework được phát triển bởi Facebook, cho phép các lập trình viên sử dụng Javascript để làm ứng dụng di động trên cả Android và iOS. Mục tiêu lớn nhất là giải quyết các bài toán hiệu năng của Hybrid App (sự kết hợp của web và ứng dụng mobile) và bài toán chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động. Một trong những lợi thế của React Native chính là chỉ cần sử dụng ngôn ngữ Javascript cho việc phát triển một ứng dụng di động hoàn chỉnh, đồng thời giải quyết được các vấn đề mà Native App thường gặp phải. Bên cạnh đó, việc sử dụng thư viện ReactJS giúp cho những ai đã có kinh nghiệm của lập trình web ReactJS có thể tiếp cận lập trình mobile một cách dễ dàng hơn so với ngôn ngữ native của từng nền tảng.

Tuy nhiên điểm yếu lớn nhất của React Native chính là hiệu năng không bằng Native App. Những ứng dụng lớn cần tối đa hiệu năng thì vẫn phải chọn native.

5. PostgreSQL

PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ-đối tượng với hơn 30 năm phát triển. Là một trong những hệ thống cơ sở dữ liệu mã nguồn mở tiên tiến nhất hiện nay. PostgreSQL sử dụng và mở rộng ngôn ngữ SQL kết hợp với nhiều tính năng mà từ đó, những dữ liệu phức tạp nhất cũng có thể scale up và được lưu trữ một cách an toàn.

6. Firebase Authentication

6.1. Giới thiệu Firebase Authentication

Firebase Authentication là một sản phẩm của Google cung cấp các dịch vụ phụ trợ cho lập trình viên. Với thư viện UI được tạo sẵn, Firebase Authentication giúp hỗ trợ xác thực bằng mật khẩu, số điện thoại và các nhà cung cấp liên kết như Google, Facebook, Twitter,...

6.2. Phương thức đăng nhập bằng Google

Phương thức đăng nhập bằng Google giúp cho người dùng không cần nhớ thông tin đăng nhập của Website khi mà chỉ sử dụng tài khoản Google đã đăng nhập trên trình duyệt, rút ngắn thời gian người dùng tạo tài khoản để sử dụng dịch vụ. Chi tiết hơn, khi chọn tài khoản để đăng nhập, Firebase Authentication sẽ trả về các thông tin cơ bản của người dùng như: tên, email, avatar URL, uid,... Từ đó, ta có thể sử dụng những thông tin này để tự động tạo tài khoản cho người dùng và khi đăng nhập lại, dùng chính uid này để lấy các thông tin đi kèm của tài khoản có trên cơ sở dữ liệu.

6.3. Control Access with Custom Claims và chức năng xác thực bằng token

Control Access with Custom Claims là một chức năng của Firebase Auth cho phép lưu các biến tùy chỉnh vào một tài khoản Google khi đăng nhập vào hệ thống, và chỉ được chỉnh sửa bằng Firebase Admin. Các thông tin đặc trưng như: giới thiệu bản thân, trang web cá nhân, tên hiển thị, avatar sẽ được lưu chung với các thông tin khác, dễ dàng hơn trong việc lưu trữ cục bộ trên trang web.

Khi đăng nhập vào hệ thống thành công, pakage *firebase/app* và *firebase/auth* sẽ được kích hoạt trả về một state chức năng. Từ đó chúng ta có thể sử dụng các hàm mà Firebase Authentication cung cấp, cụ thể là hàm lấy token xác thực getIdToken() và setCustomUserClaims(). Token sau khi được lấy sẽ được gửi qua cho server để tiến hành xác thực. Server sẽ verify IdToken để có thể lấy được các thông tin như uid và customClaims, sau đó có thể xác thực bất kỳ chức năng nào cần sử dụng xác thực tài khoản. Với đặc trưng chỉ sống được trong vòng 1 giờ, Token là một công cụ an toàn để có thể xác thực tài khoản ở cả Client và Server.

7. Prisma

Prisma là một Object Relational Mapping (ORMs) được dùng để xây dựng các máy chủ như GraphQL Server, RESTful APIs, microservice, ... Prisma đơn giản là 1 layer nằm giữa Webserver và Database. Prisma giúp chúng ta giao tiếp với database một cách dễ dàng hơn.

Prisma bao gồm ba phần chính:

+ Prisma Client: Trình tạo truy vấn an toàn và được tạo tự động cho Node.js và TypeScript.

+ Prisma Migrate: Một hệ thống di chuyển và mô hình hóa dữ liệu khai báo.

+ Prisma Studio: Một GUI để xem và chỉnh sửa dữ liệu trong cơ sở dữ liệu của bạn.

Cách truyền thống mà Web Server giao tiếp với Database là thông qua các câu lệnh SQL Query như SELECT, UPDATE hay DELETE. Giờ đây, nhờ vào các công cụ ORMs nói chung và Prisma nói riêng. Chúng tạo ra một tầng abstraction giữa

Webserver và Database. Điều này giúp cho lập trình viên dễ dàng trong việc thao tác với Database. Thay vì viết những câu lệnh SQL khô khan, có thể sai bất cứ lúc nào thì chúng ta có thể viết các hàm tương ứng.

8. Phương thức nạp tiền tự động bằng MoMo

8.1. Giới thiệu MoMo

MoMo là loại hình ví điện tử cho phép chuyển, nhận cũng như rút tiền về ngân hàng hoặc các điểm giao dịch. Với độ phủ sóng gần như cả nước, MoMo hiện nay là ví điện tử được sử dụng phổ biến nhất tại việt nam với rất nhiều tính năng mạnh mẽ.

Những đặc điểm đó là mảnh ghép hoàn hảo cho phương thức thanh toán của mô hình chợ thông tin hình ảnh.

8.2. MoMo API

MoMo Payment Platform API là giải pháp thanh toán cho các đơn vị kinh doanh, cho phép khách hàng sử dụng tài khoản Ví MoMo để thanh toán các dịch vụ trên nhiều nền tảng khác nhau: Desktop Website, Mobile Website, Mobile Application, POS, Pay In Bill, Web In App MoMo.

Để có thể sử dụng MoMo API, tài khoản cần đăng ký doanh nghiệp với MoMo. Với quy mô luận văn, tài khoản không đủ điều kiện để đăng ký doanh nghiệp nên sẽ sử dụng một bên thứ ba để thực hiện các tác vụ có liên quan đến API.

8.3. Mã giao dịch MoMo

Là một dãy các chữ số được MoMo hiển thị trong giao diện App khi thực hiện chuyển tiền thành công. Mã này là duy nhất cho mỗi giao dịch. Từ mã này, chúng ta có thể lấy các thông tin của giao dịch như số tiền, số điện thoại người gửi, tên người gửi, lời nhắn, thời gian thực hiện,… (chỉ áp dụng cho tài khoản doanh nghiệp hoặc sử dụng bên thứ ba nhưng kém an toàn hơn).

8.4. Cổng API https://momofree.apimienphi.com

Là một server do một cá nhân tạo ra để hỗ trợ những người không có đủ điều kiện đăng ký doanh nghiệp nhưng vẫn muốn sử dụng 1 số tính năng của MoMo API. Để sử dụng, chúng ta cần tải App MoMo đã được Mod của trang web, sau đó đăng nhập và sử dụng như bình thường. Hiện tại, trang web hỗ trợ các chức năng cơ bản như:

- Kiểm tra mã giao dịch: sử dụng mã giao dịch để gửi request lên server, nếu giao dịch có tồn tại trong tài khoản (được đăng nhập trong ứng dụng) thì sẽ trả về thông tin chi tiết của giao dịch. Nếu không sẽ trả về kết quả "giao dịch không tồn tại". Thông tin chi tiết như sau:

+ Mô tả:

- Method: POST.
- URL: https://momofree.apimienphi.com/api/checkTranId.

- Data Post (JSON):

{

"access_token": "he2BWRBcBZpDOsmimfSu6DXABbycBzc7n4Fp3RmbpTOpYNhOoz"

,

"tranId": "7856xxxx"

}

- access_token: là token được cấp bởi trang web
- tranId: là mã giao dịch MoMo

+ Ví dụ

- Request với mã giao dịch là 12552637945

{

"access_token": "he2BWRBcBZpDOsmimfSu6DXABbycBzc7n4Fp3RmbpTOpYNhOoz"

,

"tranId": "12552637945"

}

- Response (đã lượt bớt thông tin không quan trọng)


```
{
  "error":0,
  "data":{
    "user":"0383679476",
    "tranId":"12552637945",
    "clientTime":"2021-06-02 23:04:48",
    "ackTime":"2021-06-02 23:04:53",
    "tranType":"2018",
    "partnerId":"0337137925",
    "partnerName":"LE HOANG NHUT",
    "amount":"50000",
    "comment":"1800005",
    "status":"999",
    "desc":"Thành công",
    "extras":"
    {\"loanId\":0,\"appSendChat\":false,\"stickers\":\"\",\"themeBackground\":\"#f5fff6\",\"t
    hemeUrl\":\"https://cdn.mservice.com.vn/app/img/transfer/theme/Corona_750x260.png\",\"vpc
    _CardType\":\"SML\",\"vpc_TicketNo\":\"171.247.206.253\",\"vpc_PaymentGateway\":\"\",\"re
    ceiverMembers\":[{\"receiverNumber\":\"01683679476\",\"receiverName\":\"NGUYỄN TẤN
    ĐẠT\",\"originalAmount\":50000.0}],\"origMSource\":1,\"lixi_total\":0.0,\"lixi_count\":1,
    \"lixi_parent_id\":\"bb339a0c-0d02-46a1-ac98-
    1824b83b5f7d\",\"lixi_isFixed\":true,\"app_version\":30151,\"request_id_backend\":\"16226
    49893831_01637137925\",\"business_trans_id\":\"1622649893831_01637137925\",\"ispayment\":
    2,\"money_source\":1,\"ORIGINAL_PARTNER_ID\":\"01683679476\",\"FEE_BANK\":0.0,\"FEE_MOMO\
    ":0.0}",
    "create_time":"2021-06-02 23:07:03"
  }
```


Hình 1: Chi tiết Response

- Kiểm tra theo nội dung: Giúp kiểm tra một nội dung giao dịch có tồn tại hay không.

- Mã QR thanh toán: Mã QR để khách hàng quét và thanh toán nhanh chóng. API sẽ trả về dạng hình ảnh. Khi quét QR, sẽ đi đến trang chuyển tiền của với số điện thoại và số tiền được thêm trong URL query

- Lấy lịch sử giao dịch: Giúp lấy danh sách những giao dịch theo số điện thoại

8.5. Kết hợp API vào nạp tiền tự động

Hệ thống sử dụng chức năng "Kiểm tra mã giao dịch" để kiểm tra xem người dùng đã chuyển tiền vào tài khoản MoMo mà hệ thống cung cấp hay chưa. Nếu mã giao dịch tồn tại có nghĩa là giao dịch đã được hoàn thành. Lúc này hệ thống sẽ thực hiện cộng tiền với số tiền đã chuyển vào tài khoản và lưu mã giao dịch vào CSDL. Nếu người dùng kiểm tra mã đã tồn tại trong CSDL thì sẽ trả về kết quả "Mã giao dịch đã được sử dụng". Nếu mã giao dịch không tồn tại nghĩa là người dùng chưa chuyển tiền.

9. Các công nghệ khác có sử dụng

9.1. Tailwinds CSS

Là một tập hợp các lớp tiện ích (class) cấp thấp. Chúng có thể được sử dụng như những viên gạch để xây dựng bất kỳ loại thành phần nào. Framework bao gồm các thuộc tính CSS quan trọng nhất (nhưng có thể dễ dàng mở rộng bằng nhiều cách khác nhau). Với Tailwinds CSS, tùy biến trở nên dễ dàng hơn bao giờ hết. Framework này có tài liệu vô cùng chi tiết, bao gồm tất cả cách sử dụng chi tiết của class và chỉ dẫn nhiều cách tùy chỉnh khác nhau.

9.2. Docker

Docker là một nền tảng cho lập trình viên để phát triển, phát hành và chạy ứng dụng với các container. Nó cho phép tạo các môi trường độc lập và tách biệt để khởi chạy và phát triển ứng dụng, môi trường này được gọi là container. Khi cần deploy lên bất kỳ server nào, chỉ cần run container của Docker thì ứng dụng sẽ được khởi chạy ngay lập tức.

CHƯƠNG 2 - THIẾT KẾ GIẢI PHÁP

2.1. Phân tích thiết kế mô hình

2.1.1. Mô tả hệ thống

Thiết kế của ứng dụng chợ dữ liệu bao gồm các tác nhân và chức năng sau:

- Tác nhân khách hàng chưa đăng nhập: nhóm người dùng này có thể truy cập vào website (hoặc sử dụng App di động) và thực hiện các tác vụ như: xem danh sách các dự án đang thu thập hình ảnh, xem chi tiết yêu cầu một dự án, dùng tài khoản Google để đăng nhập.

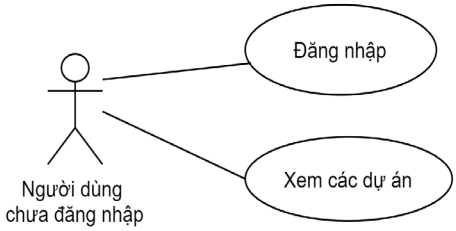
- Tác nhân người dùng đã đăng nhập: người dùng đã dùng tài khoản Google để đăng nhập, có thể là lần đầu hoặc đăng nhập lại. Tài khoản sẽ được nhớ cho lần ghé thăm website lần sau. Ngoài việc có các chức năng của người khách hàng chưa đăng nhập, nhóm người dùng này còn có các tác vụ như: tạo dự án thu thập hình ảnh hoặc thu thập hình ảnh trên bản đồ mới, gửi (bán) ảnh cho các dự án xét duyệt, xem và chỉnh sửa profile, nạp tiền tự động, xem thông tin ảnh đã gửi,...

- Tác nhân người dùng di động: Ở thời điểm hiện tại, người sử dụng di động chỉ hỗ trợ một số tính năng so với bản web. Cụ thể là: xem danh sách các dự án, xem chi tiết dự án, xem profile cá nhân, xem trạng thái những ảnh đã đóng góp. Đặc biệt, có một tính năng chỉ hoạt động trên di động đó là gửi ảnh kèm vị trí trên bản đồ, người dùng web chỉ có thể tạo dự án bản đồ mà không thể gửi ảnh cho loại dự án này.

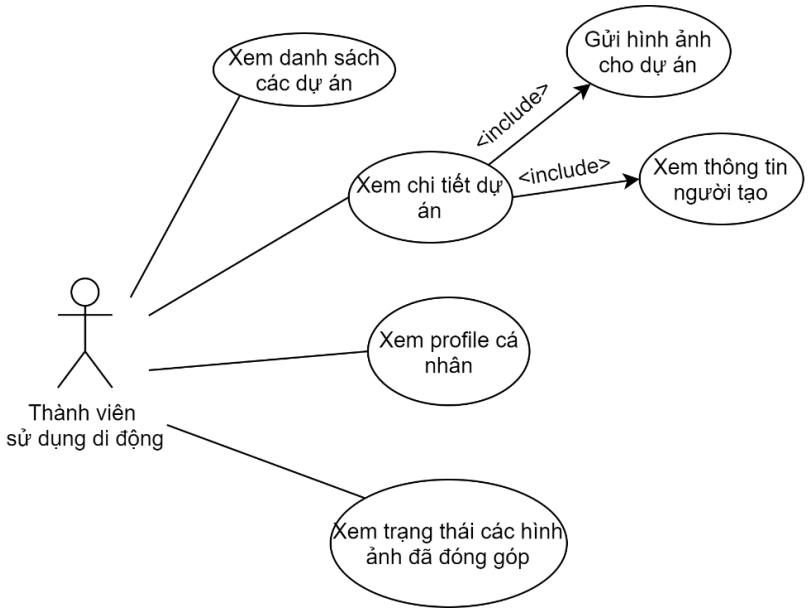
- Tác nhân Admin: là người trực tiếp quản lý hệ thống, tất cả các bài viết cần có sự phê duyệt của Admin để có thể xuất hiện trong danh sách dự án. Mọi nội dung trên website thì hầu hết Admin có thể can thiệp và chỉnh sửa. Ngoài ra, Admin sẽ là người chi trả thủ công cho các lệnh rút tiền đã tạo.

2.1.2. Thiết kế mô hình

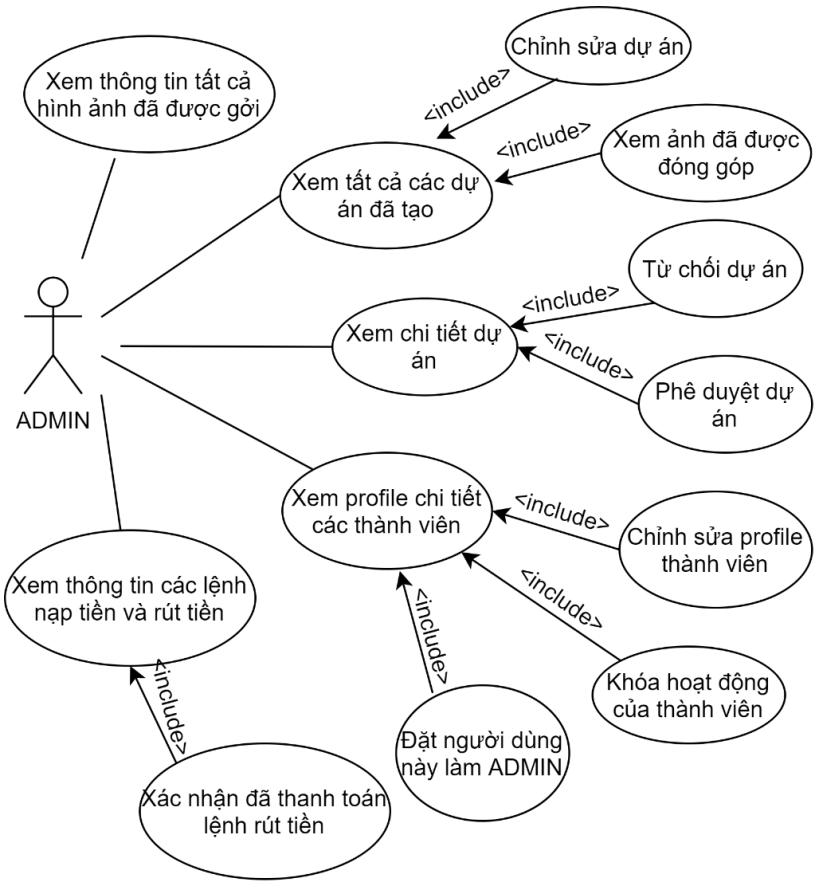
Dựa theo mô tả của hệ thống, người dùng trên website được chia thành ba đối tượng, Mỗi đối tượng người dùng sẽ có các chức năng của mình để tương tác với hệ thống. Cụ thể theo sơ đồ use case của từng tác nhân ở hình 2,3,4,5.



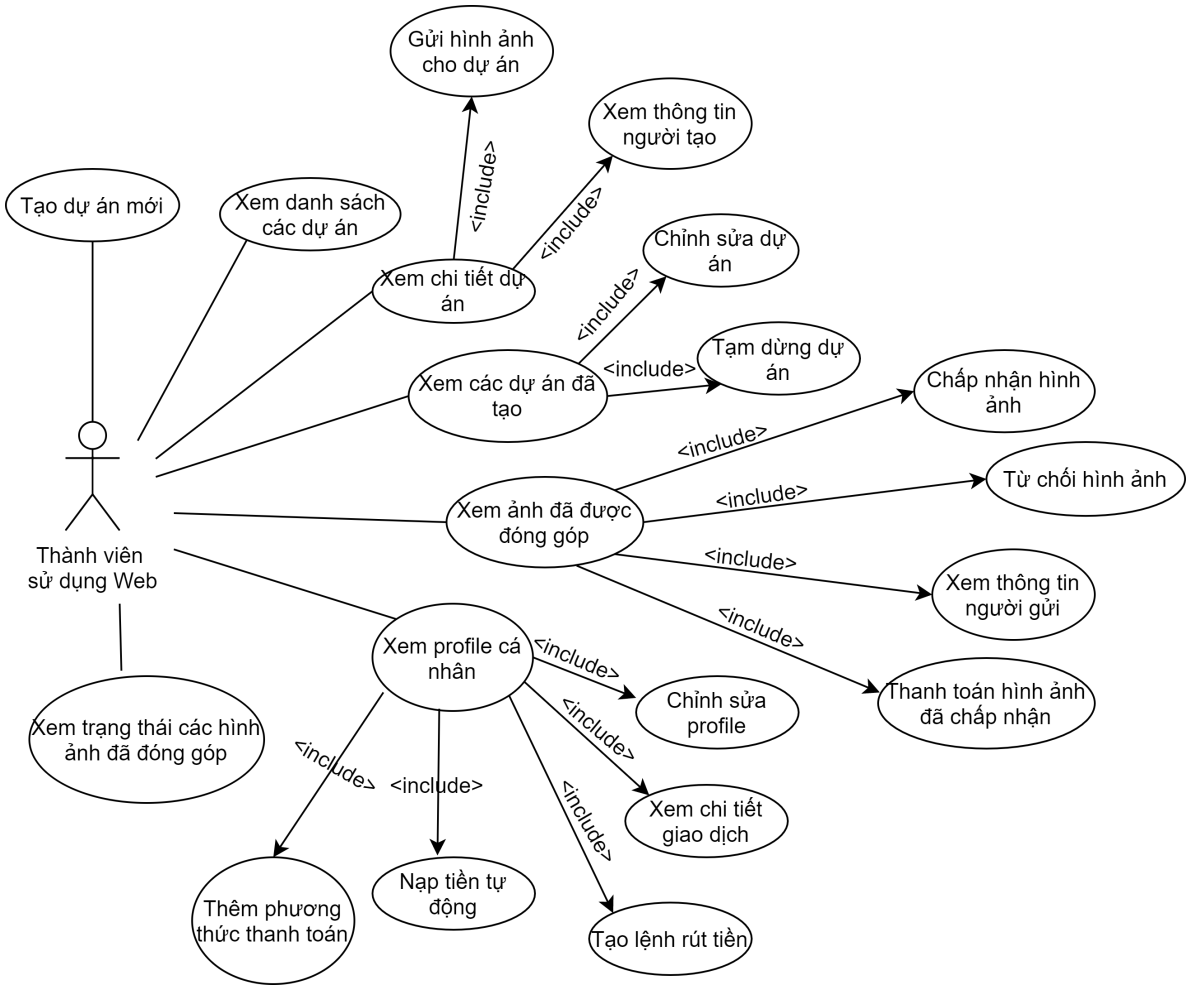
Hình 2: Sơ đồ usecase người dùng chưa đăng nhập



Hình 3: Sơ đồ usecase thành viên sử dụng di động



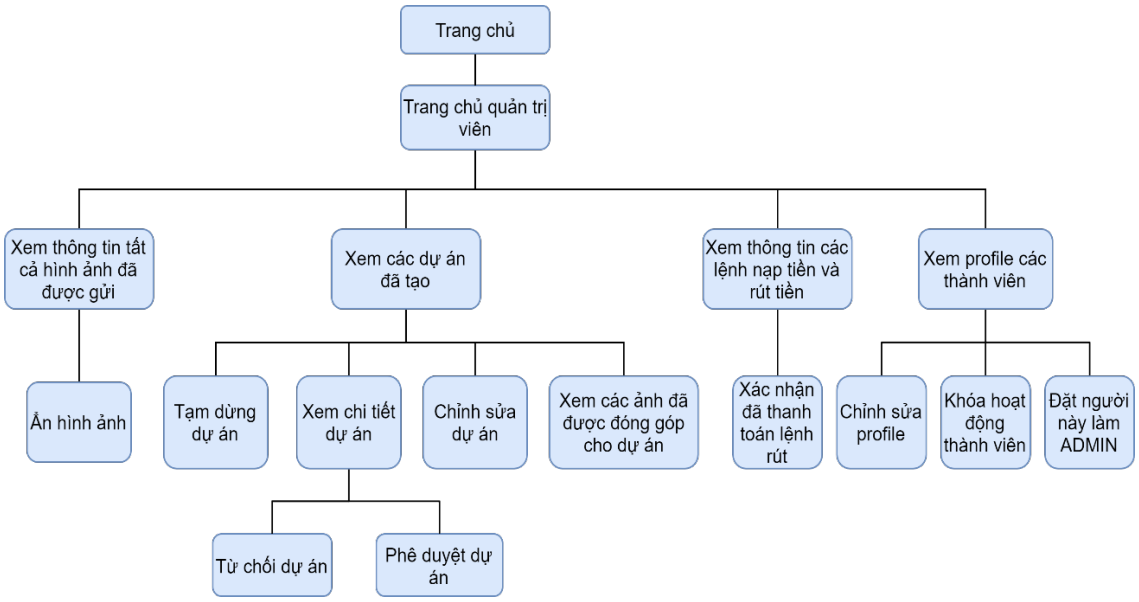
Hình 4: Sơ đồ usecase Admin



Hình 5: Sơ đồ usecase thành viên sử dụng web

2.1.3. Sơ đồ chức năng người quản trị

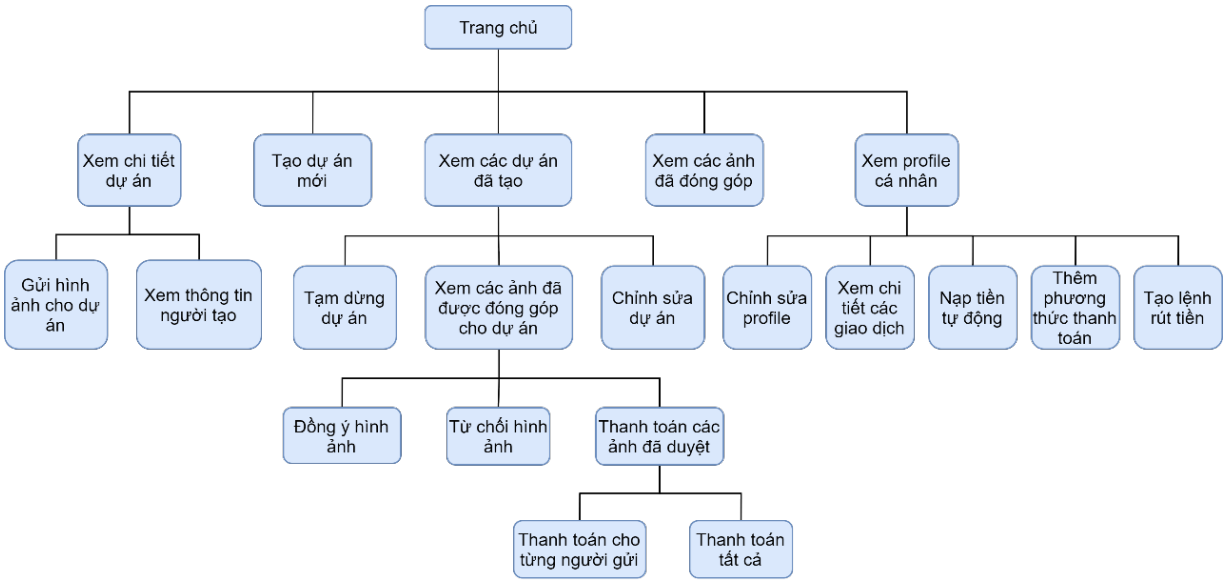
Sơ đồ thể hiện toàn cảnh các chức năng của người quản trị và cách để điều hướng đến chức năng đó, được bắt đầu từ trang chủ.



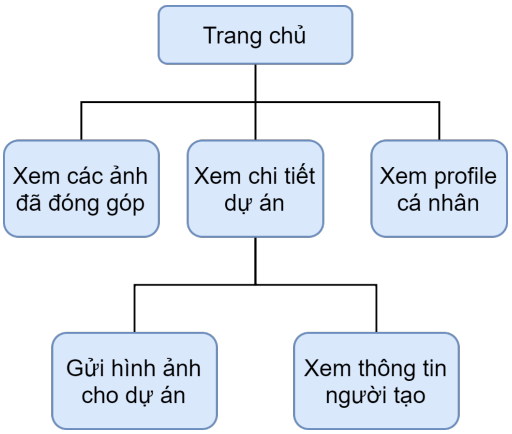
Hình 6: Sơ đồ chức năng người quản trị

2.1.4. Sơ đồ chức năng người dùng

Sơ đồ thể hiện toàn cảnh các chức năng của người dùng và cách để điều hướng đến chức năng đó. Bao gồm người dùng sử dụng web và người dùng sử dụng di động.



Hình 7: Sơ đồ chức năng người dùng web



Hình 8: Sơ đồ người dùng di động

2.2. Phân tích và thiết kế cơ sở dữ liệu

2.2.1. Xây dựng các tập thực thể

Giải thích:

- String?: có thể null
- JSON[]: mảng của 1 kiểu dữ liệu
- String (Pk): Khóa chính
- String (Fk): Khóa ngoại

2.2.1.1. Thực thể Project

- Thực thể Project chứa các thông tin chi tiết của một dự án, các thông tin hiển thị được người dùng nhập vào từ trang tạo dự án

Bảng 1: Thực thể Project

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã dự án	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	status	Trạng thái duyệt	Enum
5	complete	Trạng thái hoạt động	Enum
6	caroselImage	Hình ảnh minh họa	JSON[]
7	slug	URL dự án	String
8	type	Loại dự án	Enum
9	projectName	Tên dự án	String
10	price	Giá cho mỗi đơn vị	Float
11	maxUnit	Số lượng ảnh tối đa	Int
12	openDay	Ngày bắt đầu	Date
13	closeDay	Ngày kết thúc	Date
14	description	Ghi chú	String
15	usedFor	Mục đích sử dụng hình ảnh	String
16	requirements	Yêu cầu	String
17	contact	Địa chỉ liên lạc	String
18	authorId	ID của người tạo	String (Fk)

2.2.1.2. Thực thể User

- Thực thể User chứa các thông tin của người dùng, có 1 số thông tin hiển thị có thể chỉnh sửa được.

Bảng 2: Thực thể User

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã tài khoản	String (Pk)
2	createAt	Thời gian tạo	Date

3	updateAt	Thời gian cập nhật gần nhất	Date
4	email	Địa chỉ Email	String
5	role	Cấp bậc người dùng	Enum
6	point	Số tiền có trong tài khoản	Float
7	name	Tên người dùng	String
8	provider	Loại hình đăng nhập	Enum
9	userName	URL người dùng	String
10	userState	Trạng thái người dùng	Enum
11	photoURL	Địa chỉ avatar	String?
12	website	Website của người dùng	String?
13	bio	Giới thiệu bản thân	String?

2.2.1.3. Thực thể UserPaymentMethod

- Thực thể UserPaymentMethod chứa các phương thức thanh toán mà người dùng thêm vào, dùng để làm thông tin nhận tiền khi người dùng rút tiền. Các phương thức phải nằm trong những phương thức rút tiền khả dụng nằm ở bảng AvailableWithdrawMethod.

Bảng 3: Thực thể UserPaymentMethod

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã phương thức thanh toán	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	name	Tên người dùng	String?
5	phone	Số điện thoại	String?
6	stk	Số tài khoản ngân hàng	String?
7	paymentState	Phương thức có còn được sử dụng	Enum
8	availablePaymentMethodId	ID phương thức khả dụng	String (Fk)
9	userId	ID tài khoản	String (Fk)

2.2.1.4. Thực thể UserDepositRecorder

- Thực thể UserDepositRecorder chứa thông tin giao dịch mà người dùng đã nạp tiền vào hệ thống, hiện tại chỉ hỗ trợ MOMO.

Bảng 4: Thực thể UserDepositRecorder

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã nạp tiền	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	amount	Số tiền nạp	Float
5	userID	ID tài khoản	String (Fk)
6	availableDepositMethodId	ID phương thức nạp tiền	String (Fk)
7	momoTransactionId	ID chi tiết giao dịch MOMO	String? (Fk)

2.2.1.5. Thực thể UserWithdrawRecorder

- Thực thể UserWithdrawRecorder chứa chi tiết thông tin giao dịch của lệnh rút tiền được tạo từ người dùng.

Bảng 5: Thực thể UserWithdrawRecorder

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã rút tiền	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	amount	Số tiền nạp	Float
5	description	Ghi chú	String
6	withdrawPointState	Trạng thái rút tiền	Enum
7	userID	ID tài khoản	String (Fk)
8	availableWithdrawMethodId	ID phương thức nạp tiền	String (Fk)
9	momoTransactionId	ID chi tiết giao dịch MOMO	String? (Fk)

2.2.1.6. Thực thể MomoTransaction

- Thực thể MomoTransaction chứa chi tiết thông tin giao dịch được trả về từ API hệ thống MOMO, giao dịch được ghi vào đây là giao dịch đã thành công.

Bảng 6: Thực thể MomoTransaction

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã ID giao dịch MOMO	String (Pk)
2	createAt	Thời gian tạo	Date
3	data	Chi tiết giao dịch MOMO	JSON
4	transactionId	Mã giao dịch MOMO	String (Fk)

2.2.1.7. Thực thể AvailableWithdrawMethod

- Thực thể AvailableWithdrawMethod chứa những phương thức rút tiền được hỗ trợ.

Bảng 7: Thực thể AvailableWithdrawMethod

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã phương thức rút tiền khả dụng	String (Pk)
2	provider	Loại hình thanh toán	Enum
3	ewalletProvider	Tên nhà cung cấp ví điện tử	String?
4	bankProvider	Tên ngân hàng	String?

2.2.1.8. Thực thể AvailableDepositMethod

- Thực thể AvailableDepositMethod chứa những phương thức nạp tiền được hỗ trợ.

Bảng 8: Thực thể AvailableDepositMethod

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã phương thức nạp tiền khả dụng	String (Pk)
2	provider	Loại hình thoanh toán	Enum
3	ewalletProvider	Tên nhà cung cấp ví điện tử	String?
4	bankProvider	Tên ngân hàng	String?
5	name	Tên người nhận tiền	String
6	phone	Số điện thoại nhận tiền	String?
7	stk	Số tài khoản ngân hàng	String?

2.2.1.9. Thực thể ProjectTransactionRecorder

- Thực thể ProjectTransactionRecorder chứa thông tin giao dịch giữa các tài khoản trong với nhau, thường là giao dịch mua bán hình ảnh của một dự án.

Bảng 9: Thực thể ProjectTransactionRecorder

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã giao dịch hình ảnh	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	sumAmount	Số tiền chuyển	Float
5	senderUserId	ID tài khoản gửi	String (Fk)
6	receiverUserId	ID tài khoản nhận	String (Fk)
7	projectId	ID dự án	String (Fk)

2.2.1.10. Thực thể Media

- Thực thể Media chứa thông tin hình ảnh được người dùng gửi để xét duyệt.

Bảng 10: Thực thể Media

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã hình ảnh	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	url	URL hình ảnh có markwater	String
5	urlPaid	URL hình ảnh gốc	String
6	isApprove	Trạng thái duyệt của hình ảnh	Enum
7	paidState	Trạng thái thanh toán	Boolean
8	paidDate	Ngày thanh toán	Date?
9	project	Địa chỉ avatar	String
10	userId	ID tài khoản gửi	String (Fk)
11	projectId	ID dự án của hình ảnh	String (Fk)
12	projectTransactionRecorderId	ID record thanh toán	String? (Fk)

2.2.1.11. Thực thể MediaMap

- Thực thể MediaMap chứa thông tin thêm của thực thể Media để phù với dự án thu thập ảnh trên bản đồ.

Bảng 11: Thực thể MediaMap

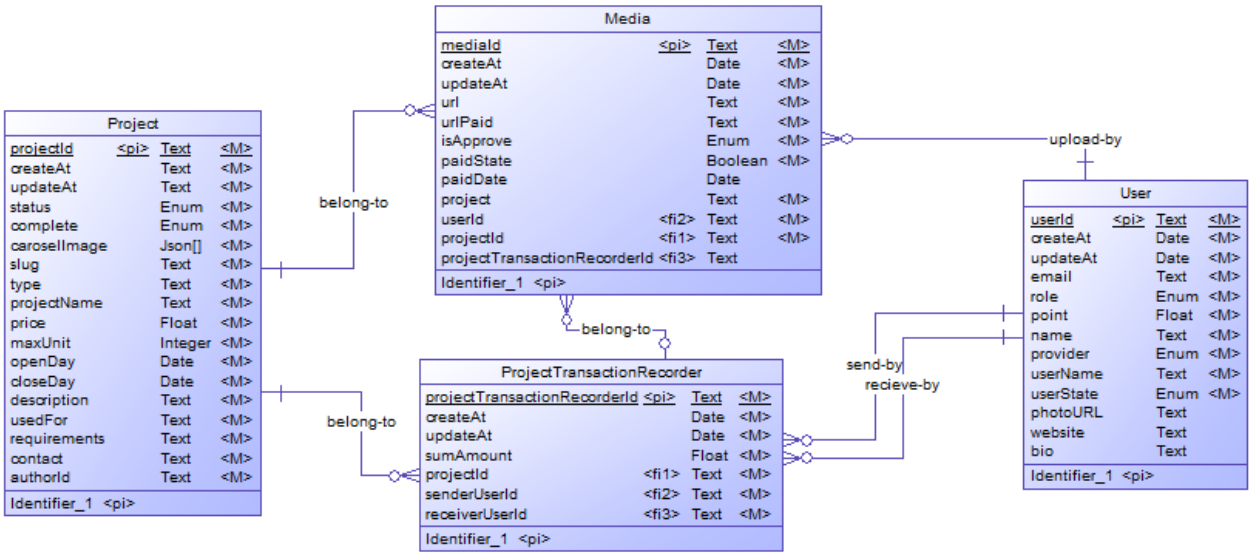
STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu
1	id	Mã hình ảnh trên bản đồ	String (Pk)
2	createAt	Thời gian tạo	Date
3	updateAt	Thời gian cập nhật gần nhất	Date
4	lat	Vĩ độ	Float
5	long	Kinh độ	Float
6	Country	Quốc gia	String?
7	City	Thành phố	String?
8	District	Huyện	String?
9	Subdistrict	Phường	String?
10	Street	Tên đường	String?
11	FullAddress	Địa chỉ đầy đủ	String?
12	mediaId	ID của hình ảnh trên bảng Media	String (Fk)

2.2.2. Mô tả các mối quan hệ trọng tâm

2.2.2.1. Mối quan hệ khi người dùng gửi ảnh cho dự án

- Khi người dùng vào chi tiết dự án (dự án thường) để gửi ảnh xét duyệt, hình ảnh sẽ được chia làm 2 copy khác nhau. Bản gốc sẽ được up thẳng lên Google Cloud Store và lấy url hình ảnh ghi vào dòng urlPaid, bản còn lại sẽ được xử lý gắn watermark và cũng up lên Google Cloud Storage, url hình ảnh được ghi vào url của bảng Media.

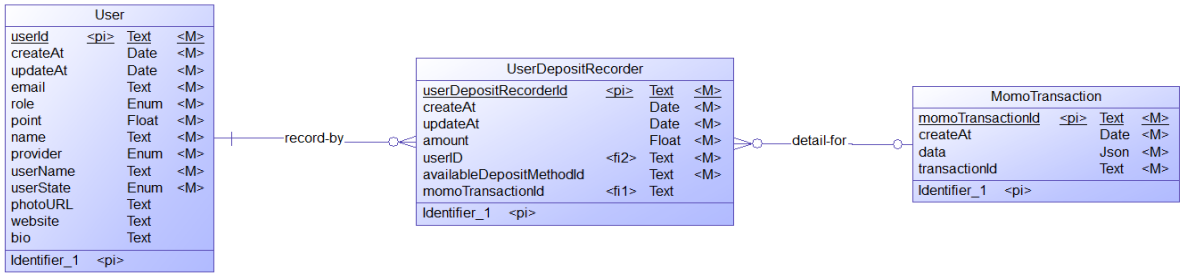
- Nếu người xét duyệt đồng ý và thanh toán hình ảnh, ProjectTransactionRecorder sẽ được thêm mới với thông tin của 2 bên. Số tiền cả 2 bên sẽ được cộng và trừ lại đúng với giá trị hình ảnh. Sau đó, hình ảnh sẽ chuyển sang trạng thái đã thanh toán và trả url hình ảnh gốc về tạo dự án. Cả quá trình thanh toán sẽ được ghi vào cơ sở dữ liệu bằng tính năng Transaction của Prisma.



Hình 9: Mối quan hệ gửi ảnh cho dự án

2.2.2.2. Mối quan hệ khi người dùng nạp tiền

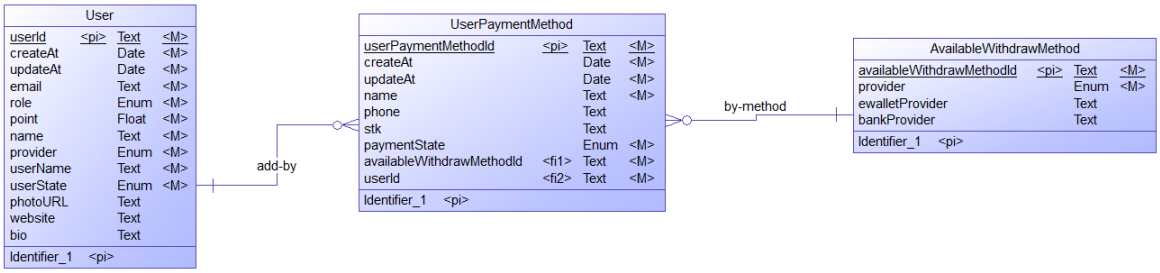
- Khi người dùng (User) sử dụng tính năng nạp tiền tự động bằng mã giao dịch MoMo. Nếu nạp tiền thành công, API MoMo sẽ trả về thông tin của giao dịch và được lưu ở MomoTransaction. Sau đó hệ thống sẽ trích thông tin số tiền nạp, kết hợp uid tài khoản gửi request để ghi vào UserDepositRecorder. Cuối cùng là cộng tiền cho User. Toàn bộ phương thức ghi vào cơ sở dữ liệu sẽ dùng tính năng Transaction của Prisma.



Hình 10: Mối quan hệ khi người dùng nạp tiền

2.2.2.3. Mối quan hệ thêm phương thức thanh toán

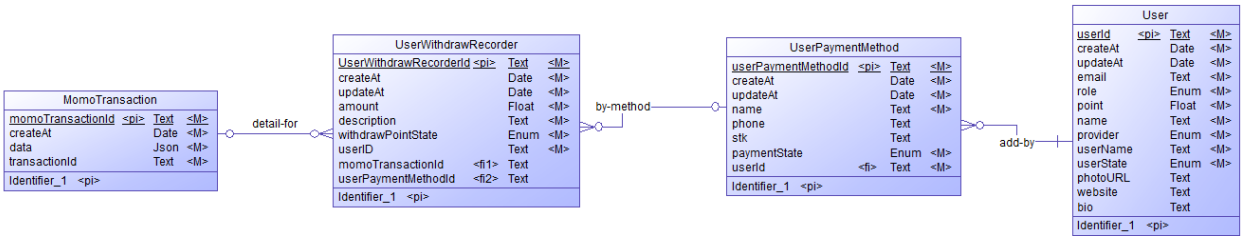
- Người dùng nếu muốn rút tiền thì phải thêm phương thức thanh toán. Các phương thức thanh toán được hỗ trợ sẽ được lấy từ bảng AvailableWithdrawMethod, quản trị có thể thêm hoặc bớt các phương thức hỗ trợ bằng cách chỉnh sửa bảng này.



Hình 11: Mối quan hệ thêm phương thức thanh toán

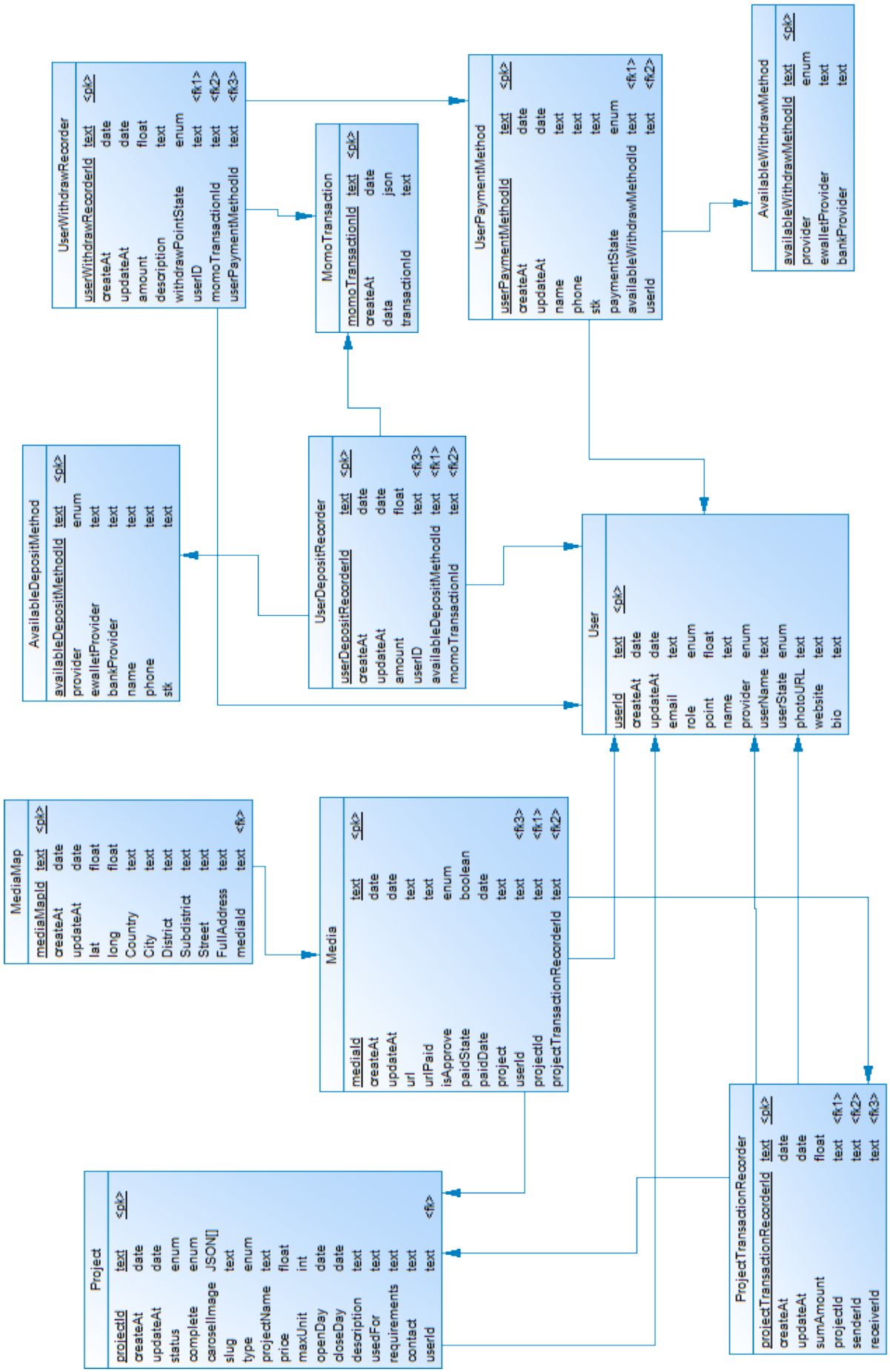
2.2.2.4. Mối quan hệ khi dùng rút tiền

- Sau khi người dùng tạo lệnh rút tiền (tiền sẽ được trừ ngay lúc tạo lệnh), thông tin từ phương thức thanh toán mà người dùng đã thêm trước đó sẽ được gửi về cho Admin. Nếu đồng ý, Admin sẽ thực hiện thanh toán thủ công và nhấn xác nhận cho lệnh thanh toán đó, nếu thực hiện thanh toán bằng MOMO thì thông tin giao dịch sẽ được thêm vào MomoTransaction. Nếu từ chối, số điểm sẽ được hoàn trả và gửi kèm lý do từ chối



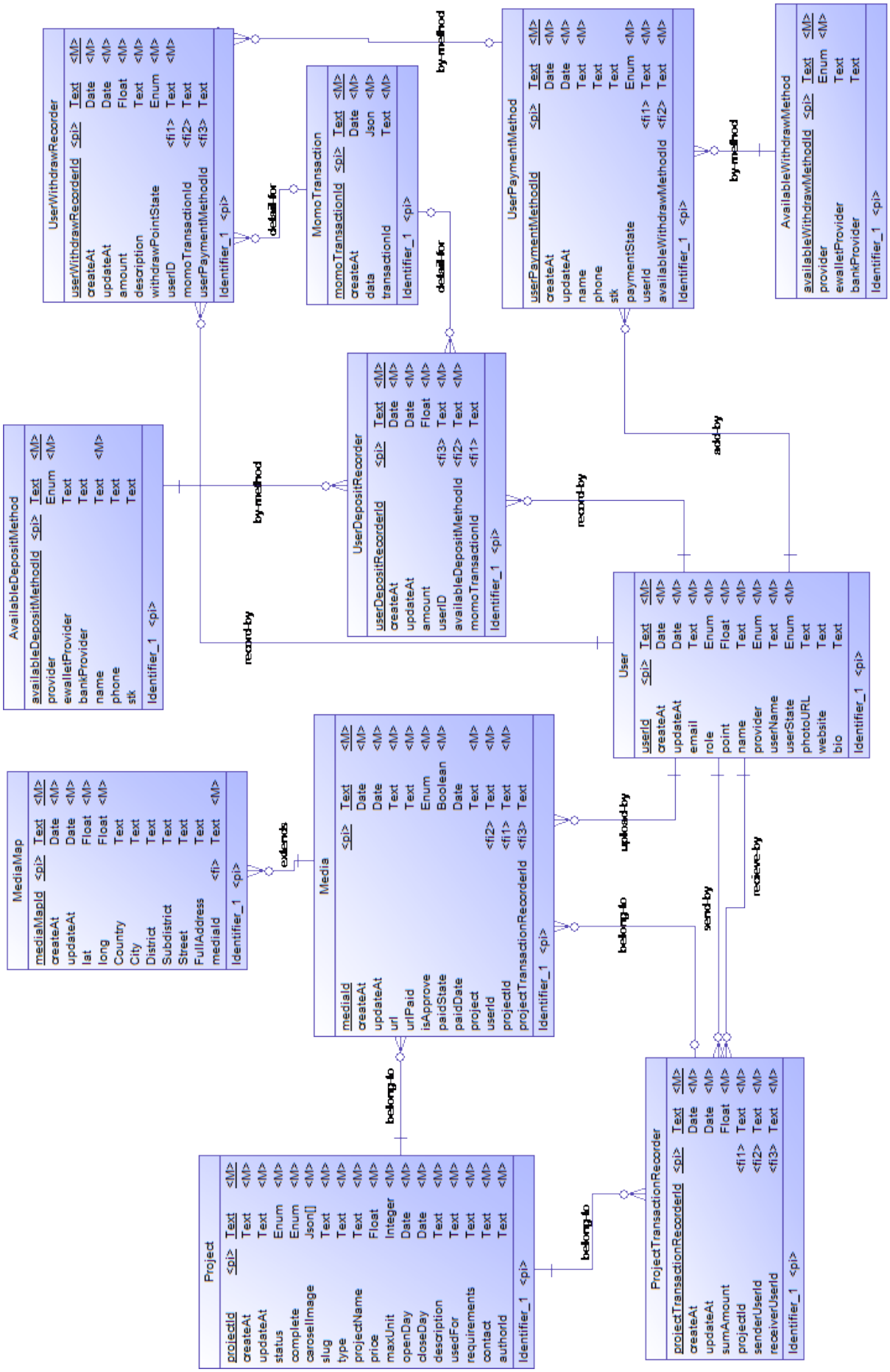
Hình 12: Mối quan hệ khi dùng rút tiền

2.2.3. Xây dựng mô hình PDM (Physical Data Model)



Hình 13: Mô hình PDM

2.2.4. Xây dựng mô hình LDM (Logical Data Model)



Hình 14: Mô hình LDM

CHƯƠNG 3 - GIAO DIỆN HỆ THỐNG VÀ CHỨC NĂNG

3.1. Giao diện cho người dùng web

3.1.1. Header

Có 3 trạng thái Header khác nhau, bao gồm:

- Header cho người dùng chưa đăng nhập

Hình 15: Header người dùng chưa đăng nhập

- Header cho người dùng đã đăng nhập, sẽ có thêm phần thông tin người dùng và điều hướng đến các chức năng khác của hệ thống

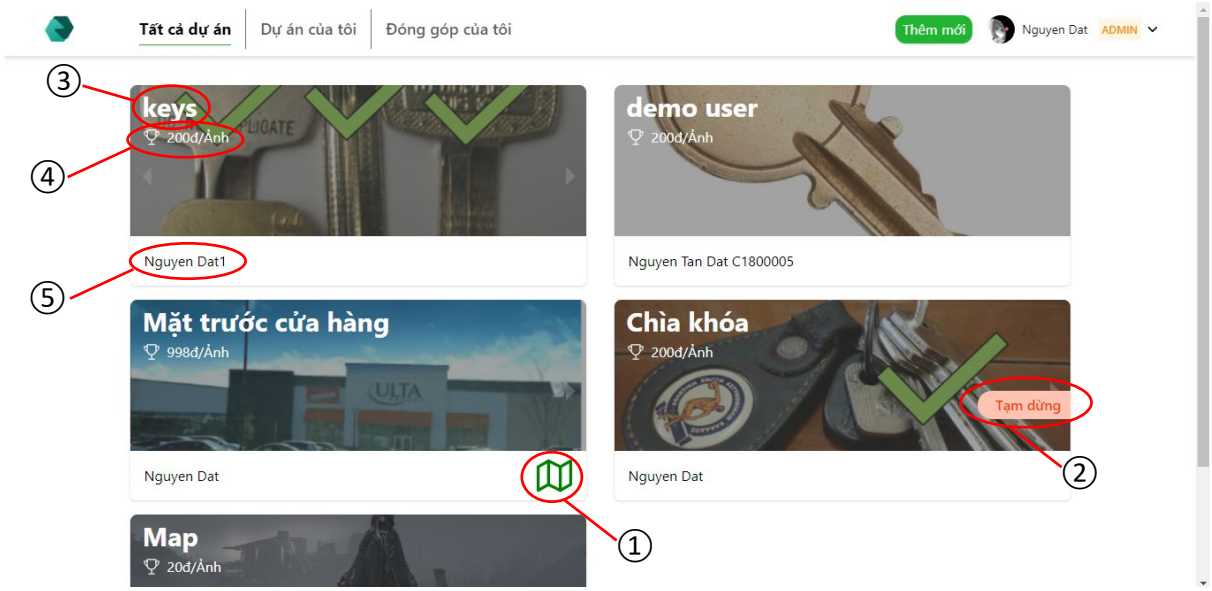
Hình 16: Header người dùng đã đăng nhập

- Header ở trang quản trị Admin, phần điều hướng đã được di chuyển thành sidebar để phù hợp cho việc quản trị.

Hình 17: Header người quản trị

3.1.2. Trang chủ

Trang chủ hiển thị thông tin sơ lược của các dự án đang hoạt động. Từ đây người dùng có thể chọn xem chi tiết 1 dự án nào đó. Người dùng *không* cần đăng nhập để có thể xem tất cả dự án.



Hình 18: Giao diện trang chủ

Chú thích:

1. Loại dự án: hiện tại có 2 loại dự án là dự án thu thập trên bản đồ và dự án thu thập ảnh bình thường. Nếu là loại dự án thu thập ảnh trên bản đồ thì sẽ hiển thị Icon như trong hình.

2. Tình trạng dự án: dự án có thể tùy chỉnh vào trạng thái tạm dừng hoạt động ở bảng hành động trong trang dự án của tôi. Ở trạng thái này, dự án vẫn xuất hiện ở trang chủ nhưng người dùng khác không thể đóng góp ảnh.

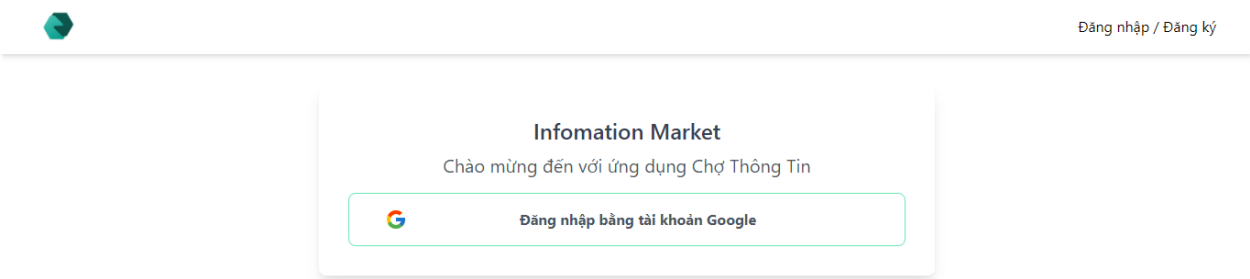
3. Tên dự án

4. Giá trên một ảnh mà dự án đưa ra

5. Người tạo dự án,

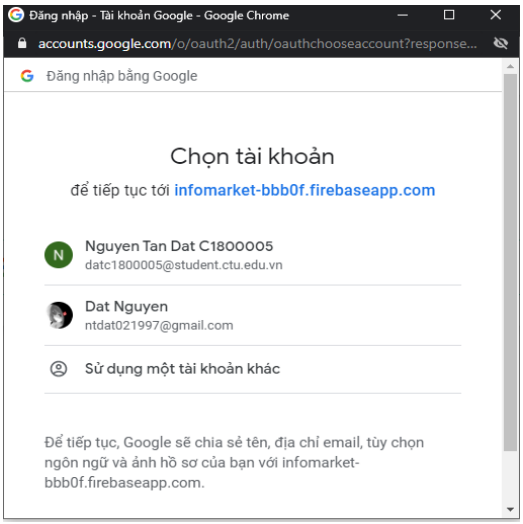
3.1.3. Trang đăng nhập

- Cho phép người dùng đăng nhập vào hệ thống. Hiện tại trang web chỉ hỗ trợ đăng nhập bằng tài khoản Google.



Hình 19: Trang đăng nhập

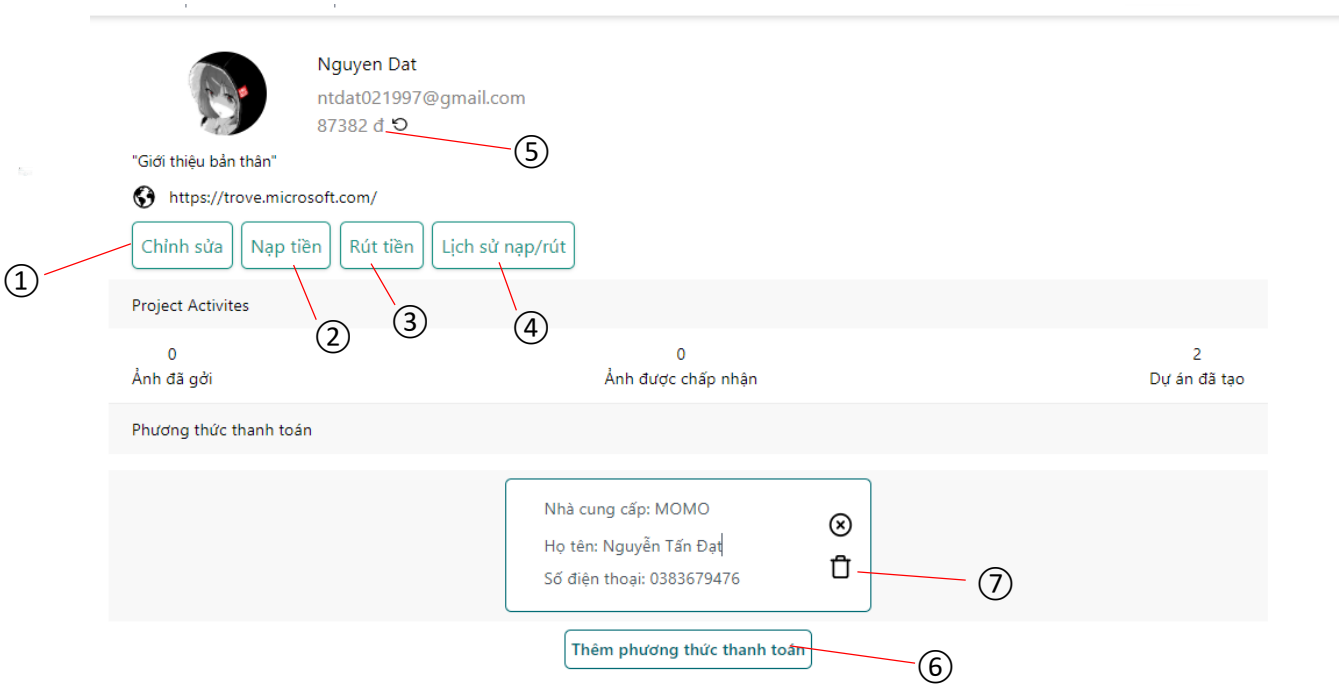
- Sau khi nhấp vào ô Đăng nhập bằng tài khoản Google, một Pop-up của Google Auth sẽ xuất hiện cho phép người chọn tài khoản Google đã lưu trên trình duyệt, nếu không có tài khoản đã lưu thì sẽ thực hiện nhập tài khoản và mật khẩu. Sau khi đăng nhập thành công, hệ thống sẽ nhận về các thông tin cơ bản như: Tên hiển thị, email, URL avatar, uid,… và các thông tin xác thực (chi tiết ở phần Firebase Auth).



Hình 20: Pop-up đăng nhập

3.1.4. Trang Profile

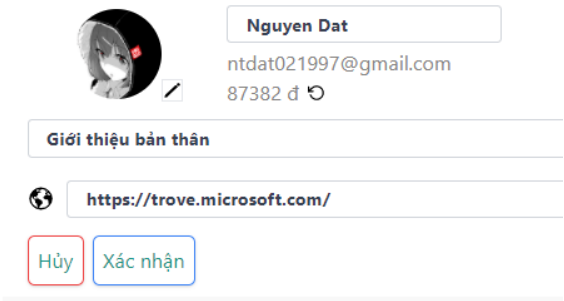
Trang Profile hiển thị các thông tin của 1 tài khoản. Tại đây có thể sử dụng một số chức năng như nạp tiền, rút tiền, chỉnh sửa trang cá nhân.



Hình 21: Trang cá nhân

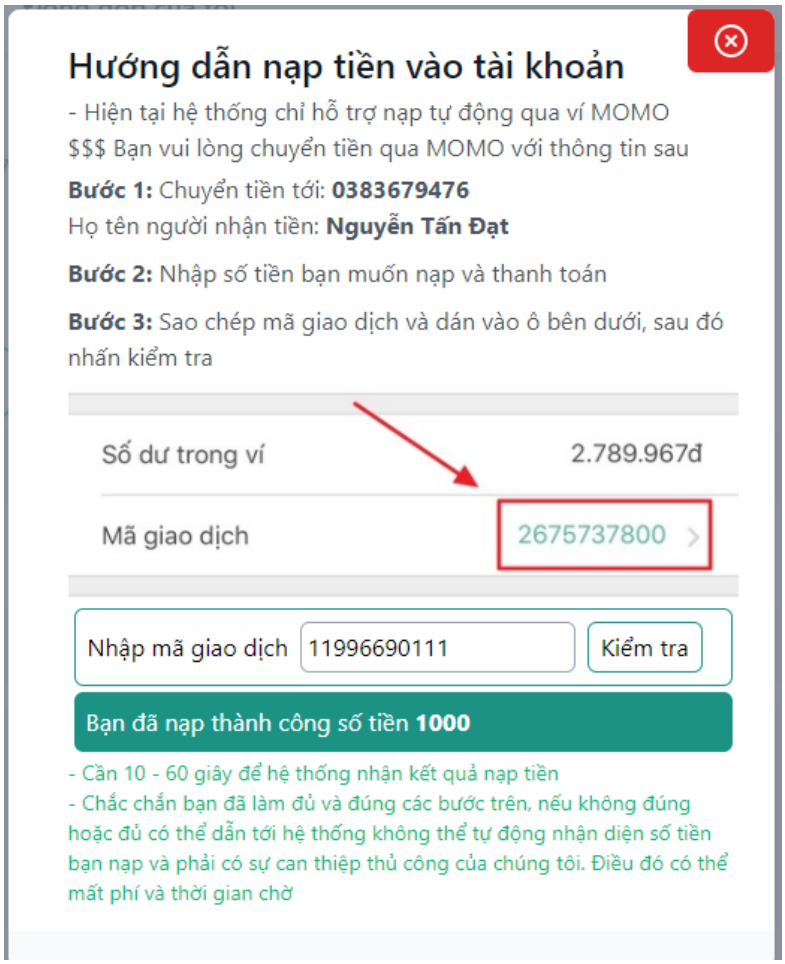
Mô tả

1. Chỉnh sửa thông tin người dùng: khi nhấp vào nút chỉnh sửa, các input sẽ được hiển thị để người dùng có thể nhập vào. Những thông tin có thể chỉnh sửa bao gồm: Tên hiển thị, giới thiệu bản thân, trang web cá nhân. Nhấp xác nhận để đồng ý sửa thông tin.



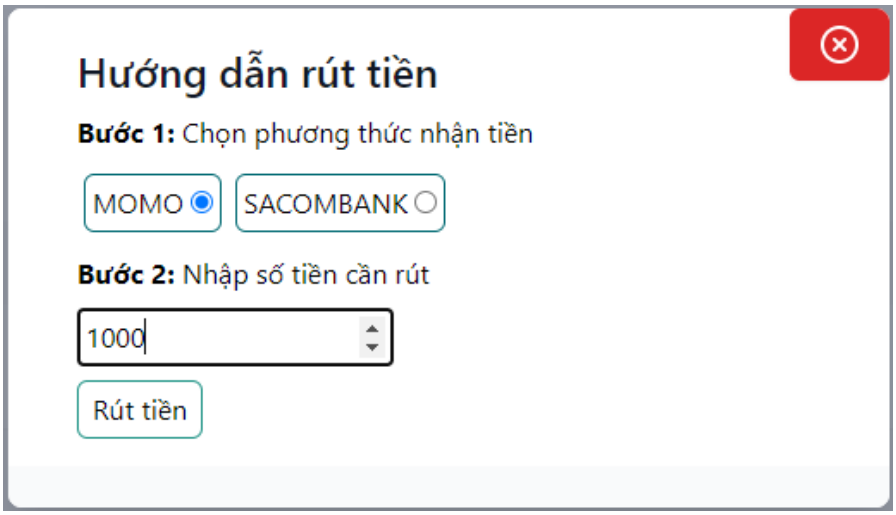
Hình 22: Chỉnh sửa trang cá nhân

2. Nạp tiền: khi nhấp vào nút nạp tiền, 1 modal sẽ xuất hiện hướng dẫn người dùng nạp tiền. Sau khi đã thực hiện chuyển tiền theo cầu, người dùng sử dụng mã giao dịch trong App MoMo để tiến hành nạp tiền.



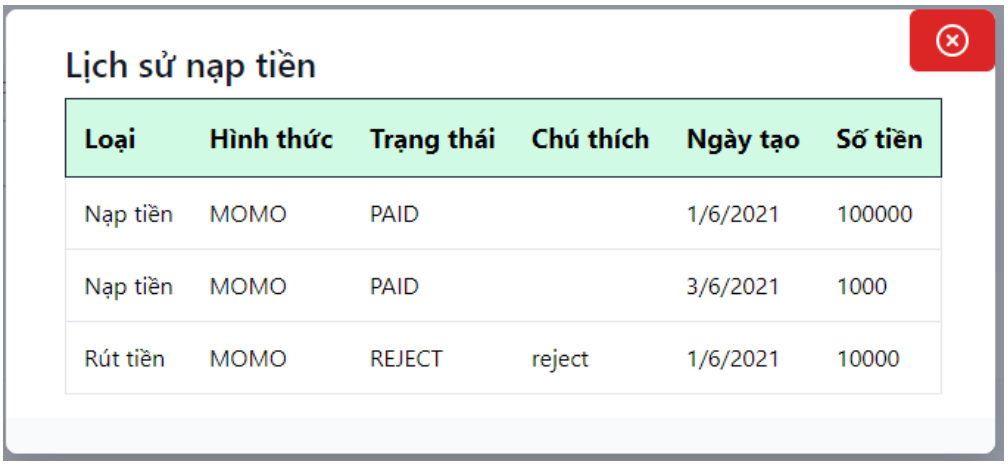
Hình 23: Hướng dẫn nạp tiền

3. Rút tiền: khi nhấp vào nút rút tiền, một modal sẽ xuất hiện yêu cầu người dùng chọn số tiền và phương thức rút tiền mà người dùng đã thêm. Sẽ không rút được nếu chưa thêm phương thức thanh toán. Sau khi nhấp rút tiền, số tiền sẽ bị trừ, lệnh rút tiền sẽ được tạo và gửi cho Admin xét duyệt. Thông tin lệnh rút tiền sẽ nằm ở modal lịch sử giao dịch (4).



Hình 24: Hướng dẫn rút tiền

4. Lịch sử nạp/rút: khi nhấp vào nút lịch sử nạp/rút, một modal sẽ xuất hiện chứa thông tin nạp tiền và rút tiền mà người dùng đã thực hiện.

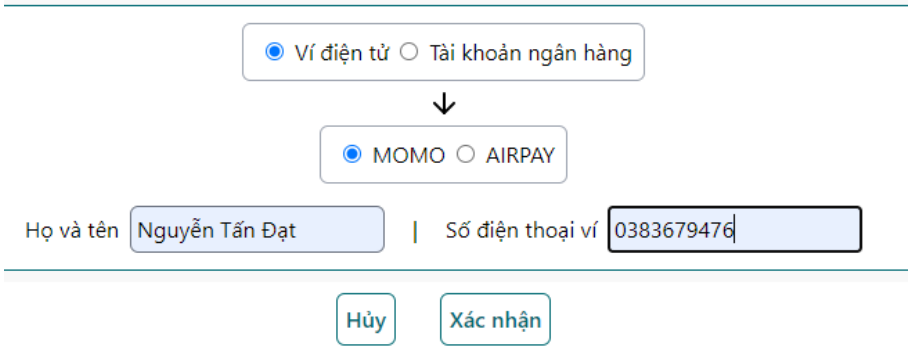


Loại	Hình thức	Trạng thái	Chú thích	Ngày tạo	Số tiền
Nạp tiền	MOMO	PAID		1/6/2021	100000
Nạp tiền	MOMO	PAID		3/6/2021	1000
Rút tiền	MOMO	REJECT	reject	1/6/2021	10000

Hình 25: Lịch sử nạp/rút

5. Số tiền đang có trong tài khoản.

6. Thêm phương thức thanh toán: dùng để thêm hình thức nhận tiền khi muốn rút tiền. Có 2 lựa chọn hiện tại là dùng ví điện tử hoặc tài khoản ngân hàng. Danh sách rút tiền khả dụng nằm ở CSDL, có thể thêm bớt tùy nhu cầu.

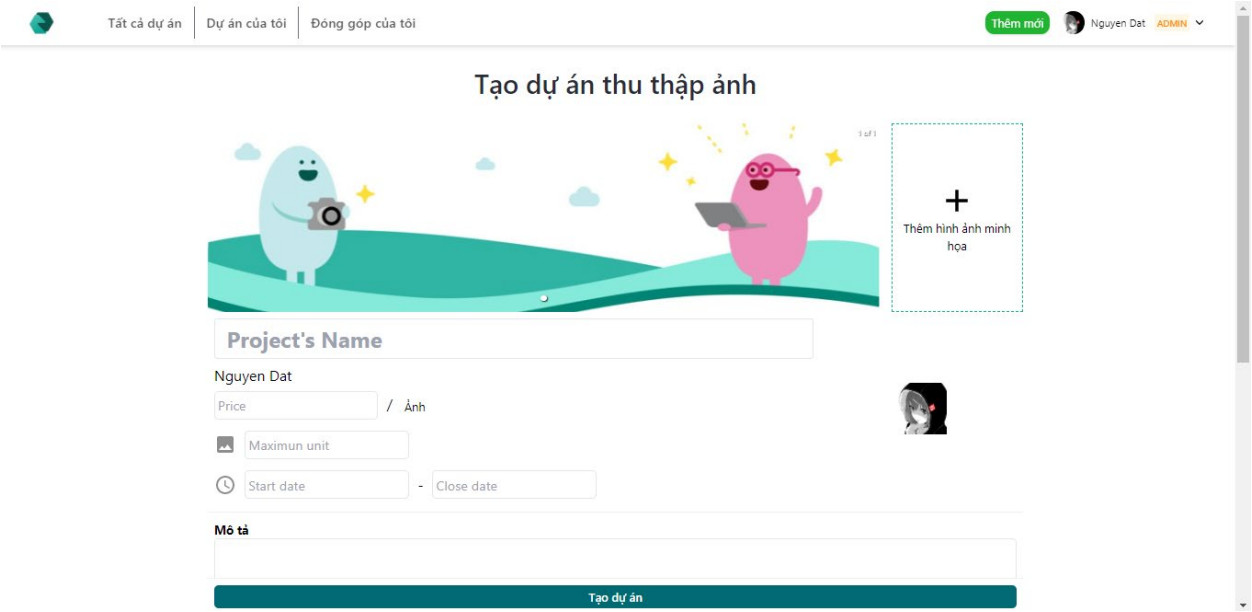


Hình 26: Thêm phương thức thanh toán

7. Xóa phương thức thanh toán: chọn để xóa 1 phương thức thanh toán đang có

3.1.5. Trang tạo mới dự án

Hiện tại, trang web có 2 loại dự án khác nhau đó là dự án thu thập ảnh và dự án thu thập ảnh trên bản đồ. Cả 2 loại đều có giao diện và thành phần giống nhau, khác nhau ở phần tiêu đề và endpoint API. Ngoài ảnh minh họa ra, tất cả các input trong trang tạo đều phải được điền vào để đảm bảo thông tin đến người dùng khác.



Hình 27: Trang tạo mới dự án

3.1.6. Trang xem chi tiết một dự án và gửi hình ảnh

- Khi nhấp vào một dự án xuất hiện ở trang chủ, người dùng sẽ đi tới trang có thể xem chi tiết về mô tả, yêu cầu, giá và các hình minh họa chi tiết. Hoặc có thể chọn Đặt câu hỏi (2) để gửi mail trực tiếp cho người tạo dự án.

Chìa khóa

$ 200đ /ảnh

500 ảnh

13/6/2021 - 11/7/2021

Nguyen Dat

Mô tả

Gửi ảnh của chìa khóa. Mỗi ảnh phải hiển thị rõ ràng các chìa khóa của bạn mà KHÔNG có người, bàn tay hoặc các vật thể khác che ảnh.

Mục đích sử dụng

Ảnh sẽ được sử dụng trong dự án AI - một ứng dụng camera biết nói, để trao quyền cho những người bị mù hoặc có thị lực kém. Ảnh cũng có thể được đóng góp vào tập dữ liệu mở để cung cấp tài nguyên cho các ứng dụng AI khác của Microsoft và các bên thứ ba. Quyền đưa ảnh của bạn vào tập dữ liệu phải được cấp phép theo Điều khoản sử dụng của Trove. Bằng cách gửi ảnh của bạn cho dự án này, bạn đồng ý cấp cho chủ sở hữu dự án quyền bổ sung đó.

Yêu cầu

1. Giữ an toàn và tuân theo tất cả các hướng dẫn cách xa xã hội trong khu vực của bạn.
2. Tối đa 10 ảnh riêng biệt. Đảm bảo tất cả chìa khóa xuất hiện trong ảnh của bạn. Mỗi bức ảnh phải hiển thị rõ ràng nhiều góc độ của đối tượng.
3. KHÔNG bao gồm bất kỳ dữ liệu cá nhân hoặc nhạy cảm nào
4. KHÔNG CÓ NGƯỜI. KHÔNG CÓ TAY trong ảnh của bạn

Liên hệ người tạo dự án

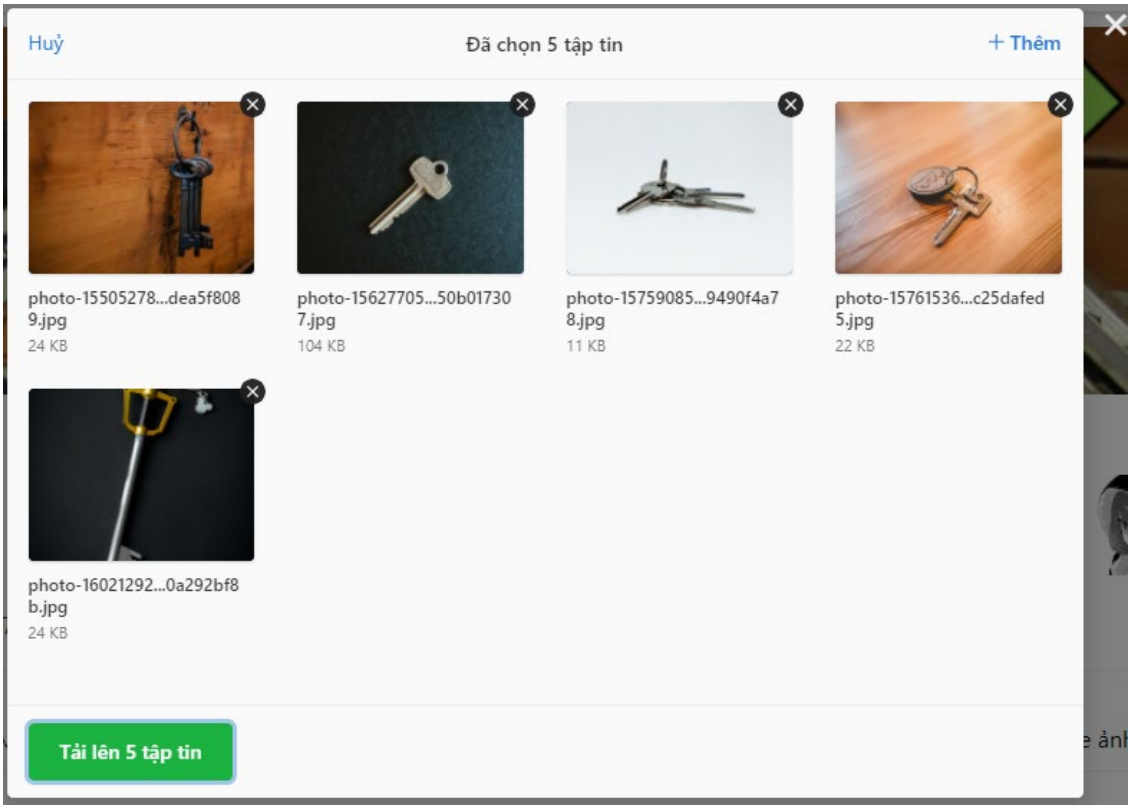
② Đặt câu hỏi

① Gửi ảnh

Hình 28: Trang chi tiết dự án

1. Gửi ảnh cho dự án: khi chọn sẽ mở một modal cho phép người dùng chọn một hoặc nhiều ảnh trong máy. Sau khi chắc chắn rằng bạn muốn gửi các ảnh đã chọn,

nhấp tải lên để bắt đầu quá trình gửi ảnh. Ảnh sau khi được gửi sẽ được chia làm bản gốc và bản có gắn watermark. Do quá trình gắn watermark được thực hiện ngay tại lúc gửi ảnh, nên sẽ mất 1 khoảng thời gian để xử lý hình ảnh. Nếu thành công hoặc thất bại thì cũng sẽ thông báo về cho người dùng.



Hình 29: Modal chọn ảnh

3.1.7. Trang dự án của tôi

Trang dự án của tôi hiển thị tất cả các dự án mà tài khoản đó đã tạo. Tại đây, ta có thể thực hiện các hành động lên dự án mình đã tạo. Ngoài ra, còn có bộ lọc để lọc theo trạng thái dự án và hiển thị số lượng dự án cho mỗi trạng thái.

Hình 30: Trang dự án của tôi

1. Tiêu đề của dự án: nhấp vào để đi đến trang xem chi tiết dự án đã tạo, có thể xem dù cho dự án có được duyệt hay chưa. Nhưng chỉ xem được các dự án của bản thân.

Hình 31: Xem dự án đã tạo

2. Tạm dừng/tiếp tục dự án: sử dụng để tạm dừng một dự án đang hoạt động hoặc tiếp tục một dự án đang tạm dừng. Dự án tạm dừng vẫn sẽ xuất hiện ở trang chủ nhưng người dùng khác không thể gửi ảnh.

3. Danh sách ảnh đã nhận của dự án: đường dẫn đến trang danh sách ảnh đã nhận được của dự án đó. Sẽ được nêu chi tiết ở phần sau.

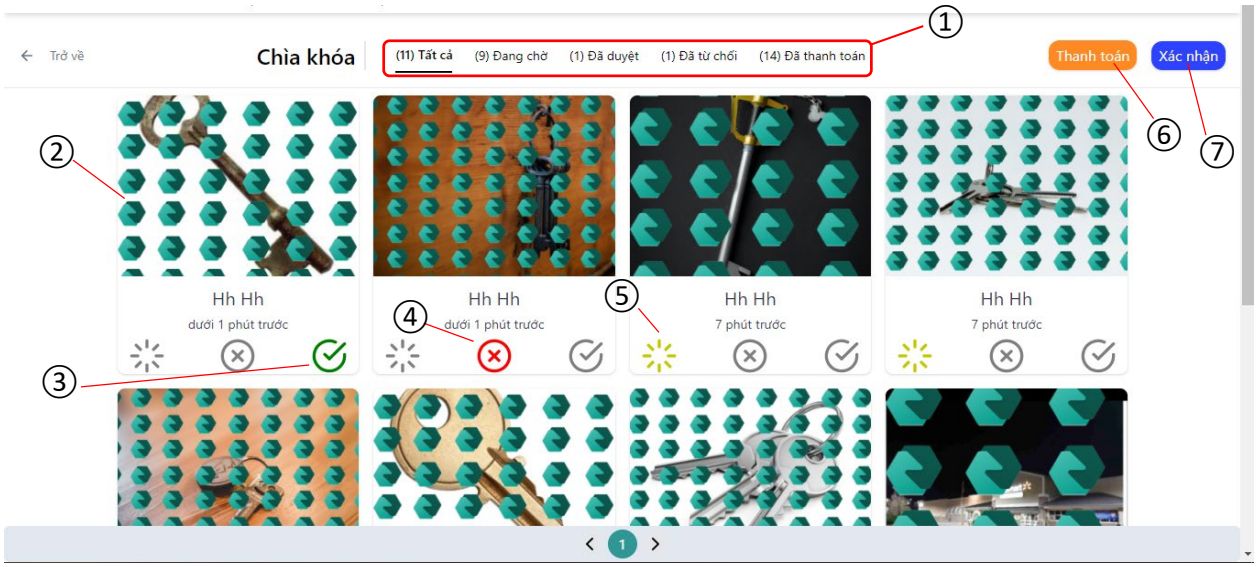
4. Chỉnh sửa dự án: đường dẫn đến trang chỉnh sửa một dự án đã chọn. Dự án sau khi cập nhật sẽ trở về trạng thái đang chờ duyệt.

Hình 32: Chỉnh sửa dự án

5. Xóa dự án: thực chất không xóa dự án mà chỉ đưa dự án vào lưu trữ. Ở trạng thái này, người dùng không thể thực hiện bất kỳ hành động nào ngoài xem chi tiết dự án. Dự án đã xóa cũng không xuất hiện trên trang chủ.

3.1.8. Trang danh sách ảnh đã nhận của dự án

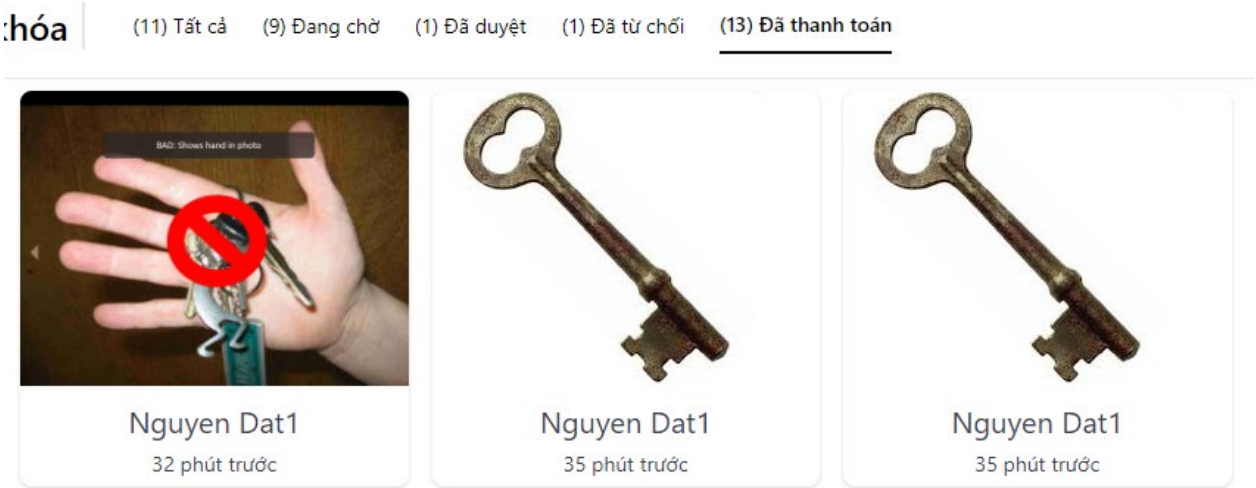
Hiển thị toàn bộ ảnh mà những người dùng khác đã đóng góp. Ảnh khi chưa được thanh toán sẽ có gắn watermark. Ảnh gốc sẽ hiển thị ở tab "Đã thanh toán" sau khi thanh toán cho người đóng góp. Ở mỗi ảnh, sẽ có những hành động cho người tạo dự án xét duyệt. Trạng thái của ảnh sẽ được cập nhật cho người gửi.



Hình 33: Xét duyệt hình ảnh

Mô tả:

1. Thanh lọc hình ảnh của trạng thái kèm số lượng của tất cả ảnh có trạng thái đó. Ảnh ở thanh "Đã thanh toán" là ảnh gốc đã được bỏ watermark.



Hình 34: Hình ảnh đã duyệt

2. Hình ảnh cho người tạo dự án xét duyệt, đã được gắn watermark. Hình ảnh gốc sẽ được hiển thị sau khi người tạo dự án thanh toán cho ảnh. Ngoài ra, người tạo dự án cũng có thể nhấp vào hình ảnh để nhìn toàn màn hình mỗi ảnh.

3,4,5. Mỗi hình ảnh mang một trạng thái duyệt riêng biệt, nếu ảnh ở trạng thái nào thì icon ở trạng thái đó sẽ có màu. Người xét duyệt có thể nhấn vào mỗi nút để thay đổi trạng thái cho hình ảnh và nhấn xác nhận để lưu thay đổi.

6. Đường dẫn đến trang thanh toán hình ảnh. Sẽ giới thiệu chi tiết ở phần sau.

7. Sau khi thay đổi trạng thái của bất kỳ ảnh nào, nhấn vào nút "Xác nhận" này để lưu thay đổi. Nếu thay đổi trạng thái ảnh mà không xác nhận, các ảnh sẽ giữ nguyên

trạng thái trước đó. Một thông báo với số lượng ảnh đã thay đổi trạng thái sẽ xuất hiện nếu quá trình thay đổi thành công.

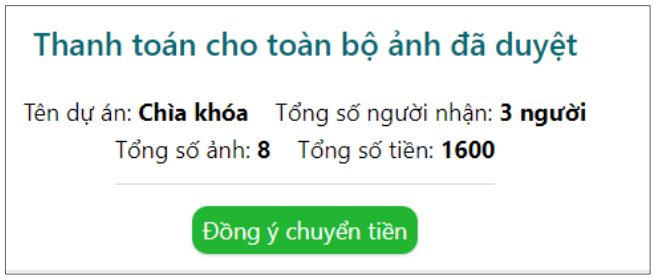
3.1.9. Trang thanh toán hình ảnh

Tất cả hình ảnh sau khi được đồng ý bởi người xét duyệt sẽ xuất hiện ở trang thanh toán hình ảnh (không bao gồm các ảnh đã được thanh toán). Các ảnh đã được gom nhóm lại theo tài khoản cùng lựa chọn thanh toán cho từng người. Mọi giao dịch thanh toán có thay đổi trên CSDL đều được sử dụng tính năng Transaction của Prisma

Hình 35: Thanh toán hình ảnh

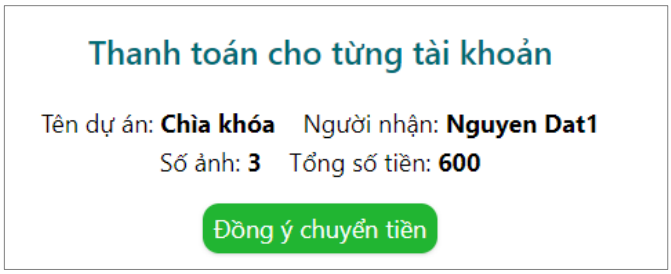
Mô tả:

1. Thanh toán tất cả: thực hiện hành động thanh toán tất cả các ảnh đã được duyệt ở thời điểm hiện tại. Nếu giao dịch thành công, số tiền của người tạo dự án sẽ được chia đúng với số ảnh của từng tài khoản.



Hình 36: Thông tin thanh toán hàng loạt

2. Thanh toán cho từng tài khoản riêng: thực hiện hành động thanh toán cho các ảnh đã được duyệt của tài khoản đã chọn.

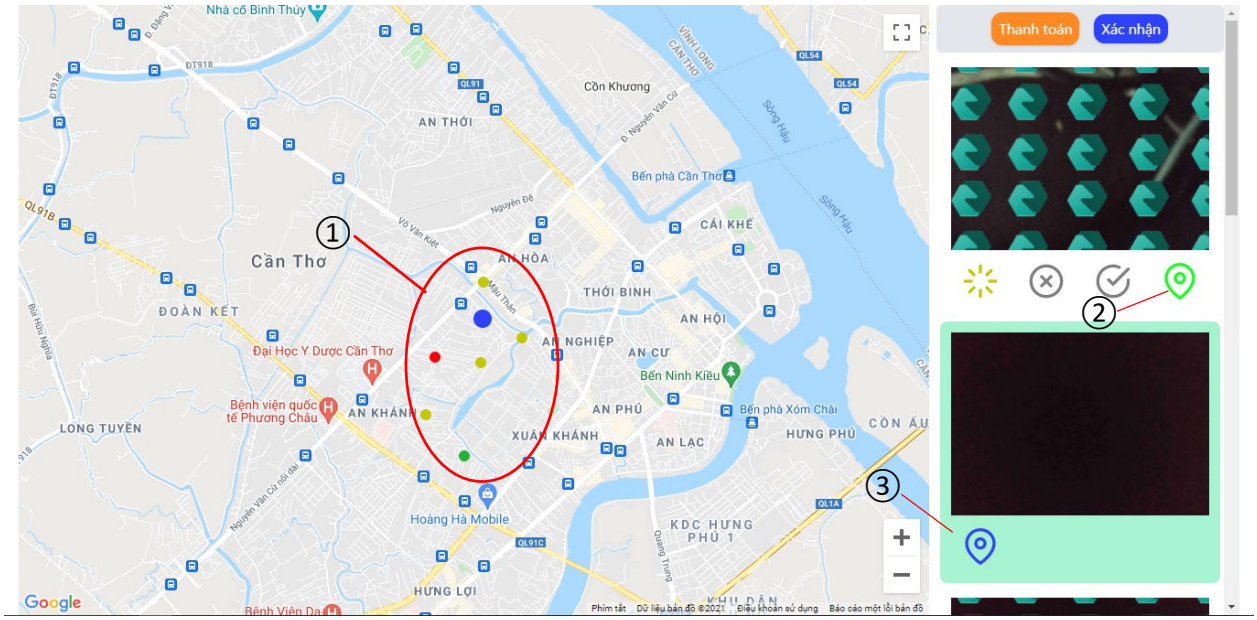


Hình 37: Thông tin thanh toán từng thành viên

3. Xem hình ảnh trước khi thanh toán: khi nhấn vào có thể xem danh sách các hình ảnh để thanh toán.

3.1.10. Trang duyệt hình ảnh trên bản đồ

Hiển thị tất cả ảnh mà người dùng đã gửi cho dự án, được thể hiện qua các chấm nằm ở những vị trí trên bản đồ. Các ảnh được chụp bằng thiết bị di động và vị trí được lưu tại vị trí mà người gửi chụp ảnh. Các tùy chọn duyệt ảnh tương tự như dự án thường.



Hình 38: Hiển thị thông tin hình ảnh trên bản đồ

Mô tả:

1. Hiển thị vị trí:

- Các chấm với các màu khác nhau thể hiện trạng thái duyệt của ảnh đó ở vị trí đó. Chi tiết trạng thái của mỗi màu:

+ : hình ảnh đang đợi duyệt.

+ : hình ảnh đã bị từ chối.

+ : hình ảnh đã được duyệt.

+ : hình ảnh đã thanh toán.

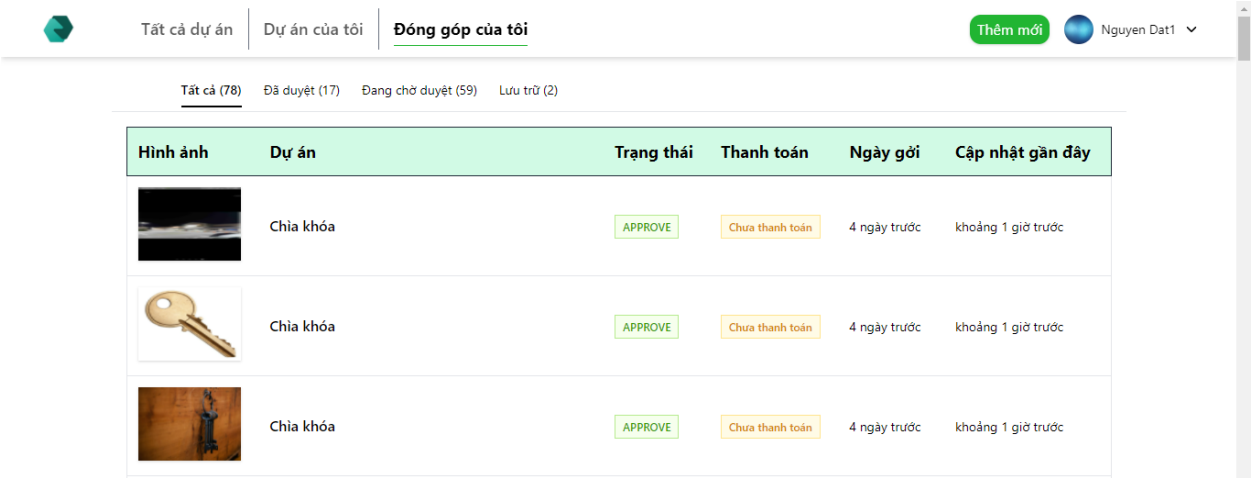
- Người dùng có thể tương tác với các chấm bằng cách hover hoặc nhấn chọn, khi đó các chấm sẽ có kích thước to hơn các chấm còn lại. Ngoài ra, background của hình ảnh ở sidebar cũng khác màu so với các hình còn lại. Tương tự, khi hover hoặc nhấn chọn các ảnh bên sidebar, chấm trên bản đồ của hình ảnh đó sẽ to hơn các chấm còn lại

2. Center vị trí: khi nhấn vào biểu tượng này, vị trí chấm màu của hình ảnh sẽ nằm ở trung tâm của bản đồ.

3: Xem ảnh của vị trí: khi nhấn vào một chấm màu trên bản đồ, hình ảnh của vị trí đó sẽ được ScrollTo và background của hình ảnh đó trở nên sáng màu.

3.1.11. Trang đóng góp của tôi

Hiển thị tất cả hình ảnh mà bản thân tài khoản đã đóng góp, kèm theo đó là các trạng thái được cập nhật từ người người tạo dự án đó.



Hình 39: Trang đóng góp của tôi

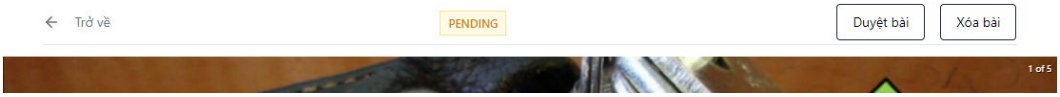
3.2. Giao diện cho người quản trị

3.2.1. Trang quản lý dự án

Hiển thị tất cả các dự án đang tồn tại trên hệ thống. Người quản trị có toàn quyền đối với các dự án này như: xóa, chỉnh sửa, tạm dừng, duyệt dự án, xem ảnh đóng góp,…

Hình 40: Trang quản lý dự án

1. Đường dẫn đến trang preview và duyệt dự án.



Hình 41: Duyệt dự án

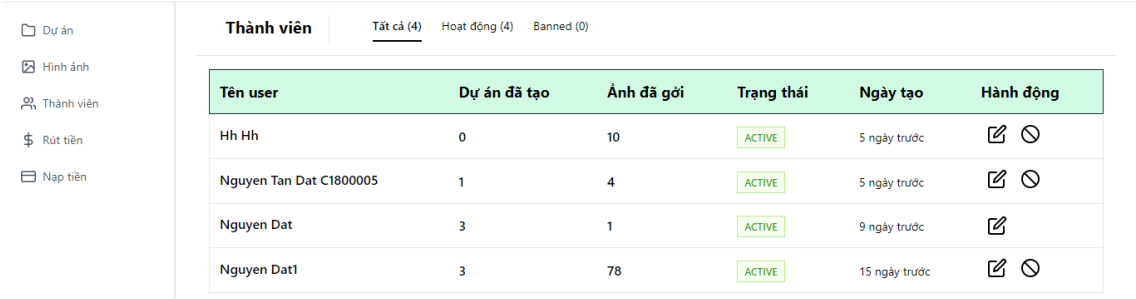
3.2.2. Trang quản lý ảnh đóng góp

Bao gồm tất cả ảnh đã được đóng góp bởi người dùng.

Hình 42: Trang quản lý hình ảnh đóng góp

3.2.3. Trang quản lý thành viên

Tất cả các tài khoản sau khi đăng nhập bằng Google sẽ được lưu thông tin và hiển thị tại trang này. Người quản trị có quyền chỉnh sửa, tạm dừng hoạt động của một tài khoản hoặc đặt người dùng này làm người quản trị. Tài khoản quản trị không thể tạm dừng.



Hình 43: Trang quản lý thành viên

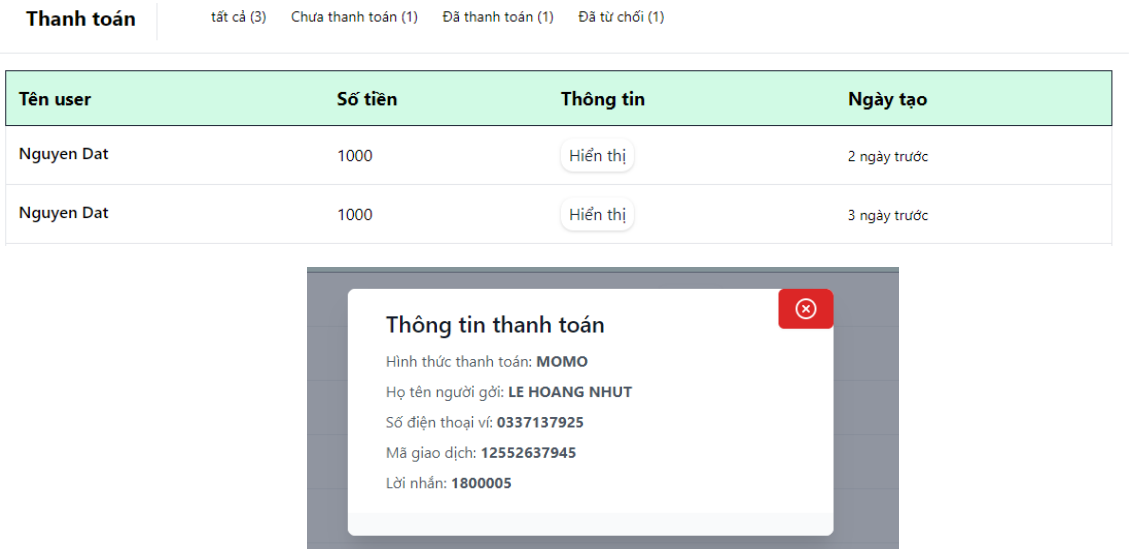
3.2.4. Trang quản lý nạp và rút tiền

Hiển thị tất cả các lệnh nạp và rút của thành viên:

- Đối với các lệnh rút tiền, người quản trị sẽ chuyển tiền thủ công cho thành viên. Kèm theo đó là chức năng ghi chú để thông báo cho người dùng lý do từ chối thanh toán.

Hình 44: Trang quản lý nạp và rút tiền

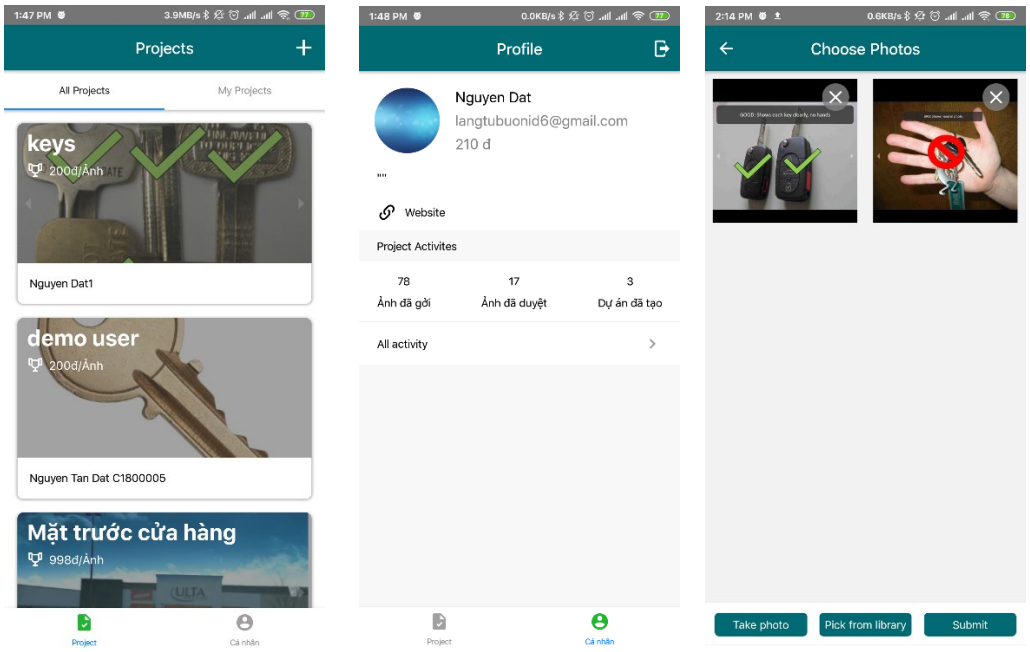
- Đối với các lệnh nạp tiền, người quản trị có quyền xem thông tin căn bản của tài khoản chuyển tiền. Hiện tại chỉ hỗ trợ xem thông tin ví MoMo



Hình 45: Hiển thị các lệnh nạp và thông tin người nạp

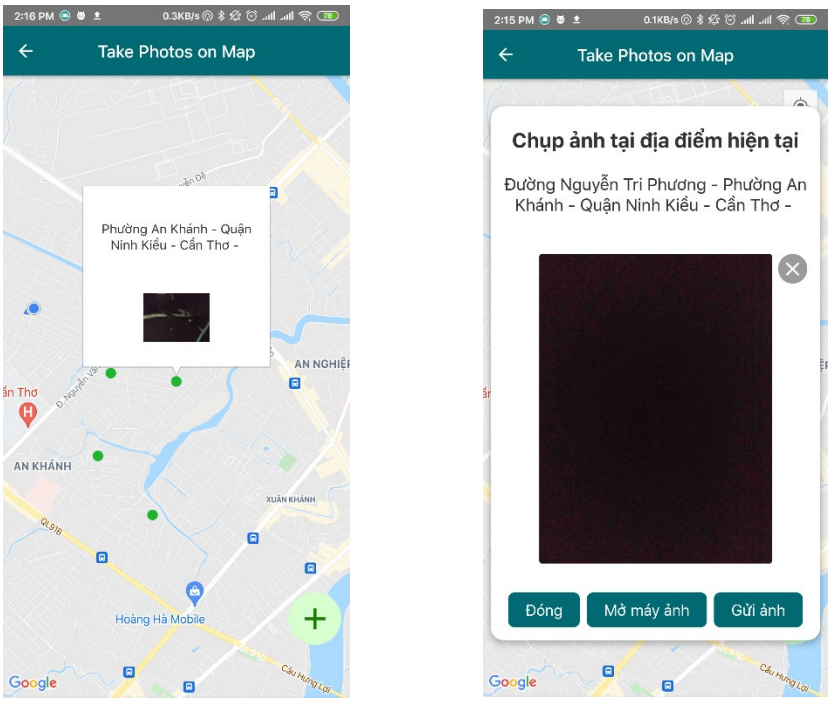
3.3. Giao diện cho người dùng di động

- Hiện tại, ứng dụng di động chỉ hỗ trợ các chức năng căn bản cho việc đăng nhập và gửi ảnh lên dự án. Kèm theo đó là chức năng gửi ảnh trên bản đồ. Một số hình ảnh về giao diện ứng dụng di động:



Hình 46: Giao diện trang chủ, trang cá nhân và trang chọn hình ảnh

- Mô tả chức năng gửi ảnh trên bản đồ: khi thành viên di duyển đến một vị trí nào đó và bấm thêm ảnh, hệ thống sẽ tự nhận diện địa chỉ của vị trí hiện tại và hiển thị lên màn hình. Nếu người dùng chụp ảnh và đồng ý gửi ảnh, hình ảnh kèm vị trí sẽ được gửi lên hệ thống. Từ đó, người tạo dự án có thể xem lại hình ảnh được hiển thị trên bản đồ.



Hình 47: Giao diện chụp ảnh trên bản đồ

CHƯƠNG 4 - ĐÁNH GIÁ VÀ KIỂM THỬ

4.1. Mục tiêu

- Nhằm xác định các lỗi trong trường hợp có thể xảy ra của ứng dụng.

- Bao gồm các hoạt động đảm bảo rằng các tính năng của hệ thống thực hiện đúng yêu cầu chức năng đã được đặt tả.

- Đảm bảo tính hoàn thiện của ứng trước khi cho sản phẩm chạy chính thức.

4.2. Phạm vi kiểm thử

Các trường hợp kiểm thử tương ứng với một số chức năng được mô tả trong tài liệu này.

4.3. Quy trình kiểm thử

Quá trình kiểm thử hệ thống sẽ được tiến hành theo các bước sau:

- Lập kế hoạch kiểm thử.

- Tiến hành kiểm thử.

- Báo cáo kết quả.

4.4. Môi trường kiểm thử

4.4.1. Môi trường chạy ứng dụng

- Laptop cá nhân đã được cài đặt đủ môi trường để chạy "Ứng dụng mua bán thông tin" và có kết nối Internet.

- Điện thoại đã cài ứng dụng và sử dụng Router wifi để kết nối với server chạy trên laptop cá nhân.

4.4.2. Chi tiết cấu hình máy chạy ứng dụng

- Laptop cá nhân:

  + Tên thiết bị: Dell Inspiron 15-5548.

  + CPU: Chip Intel Core i7 – 5500U, 2,6GHZ.

  + RAM: 8GB.

  + Hệ điều hành: Windows 10.

  + Trình duyệt Chrome.

- Điện thoại di động cá nhân:

  + Tên thiết bị: Xiaomi Note 5.

  + CPU: Snapdragon 636.

  + RAM: 4GB.

+ Hệ điều hành: Android 8.1.0.

+ Màn hình: 5.99", FHD+, IPS LCD, 1080 x 2160 Pixel.

4.5. Các chức năng kiểm thử

Một số chức năng sau sẽ được kiểm thử:

- Đăng nhập.
- Nạp tiền vào tài khoản.
- Tạo lệnh rút tiền.
- Tạo dự án mới.
- Gửi ảnh cho dự án.
- Gửi ảnh trên bản đồ trên thiết bị di động.
- Thanh toán ảnh đã duyệt.

4.6. Một số chức năng không được kiểm thử

Những chức năng không được kiểm thử là những chức năng đơn giản, không có nhiều trường hợp xảy ra hoặc các chức năng đó tương tự như các chức năng đã được kiểm thử.

4.7. Tiêu chí kiểm thử

Tiêu chí kiểm thử thành công là kết quả thực hiện chức năng đúng với mong đợi, phù hợp với đặc tả yêu cầu.

Tiêu chí kiểm thử thất bại là kết quả không như mong đợi, xuất hiện lỗi, không phù hợp với đặc tả yêu cầu.

4.8. Kiểm thử chức năng

4.8.1. Đăng nhập

- Mục tiêu: Kiểm thử chức năng đăng nhập có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 12: Kiểm thử chức năng đăng nhập

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH1_01	Trình duyệt chưa đăng nhập tài khoản Google, tiến hành nhập đúng tài khoản và mật khẩu	Đăng nhập vào hệ thống thành công	Đăng nhập thành công
TH1_02	Trình duyệt đã đăng nhập vào tài khoản Google trước đó, thực hiện một chọn tài khoản	Đăng nhập vào hệ thống thành công	Đăng nhập thành công

TH1_03	Trình duyệt chưa đăng nhập tài khoản Google, tiến hành nhập sai email và mật khẩu	Lỗi sai tên tài khoản hoặc mật khẩu	Đăng nhập thất bại
TH1_04	Đăng nhập vào tài khoản đã bị khóa bởi Admin	Tự thoát ra khỏi tài khoản	Đăng nhập thất bại
TH1_05	Tắt pop-up đăng nhập của Google khi nhấn đăng nhập	Chưa đăng nhập vào hệ thống	Đăng nhập thất bại

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng đăng nhập hoạt động tốt.

4.8.2. Nạp tiền vào tài khoản

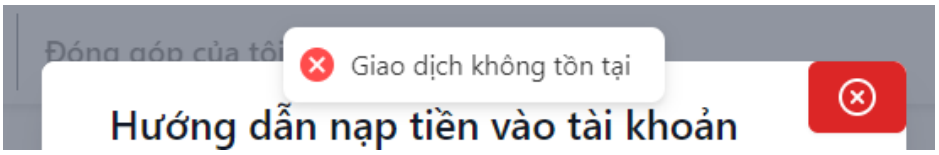
- Mục tiêu: Kiểm thử chức năng nạp tiền có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 13: Kiểm thử nạp tiền vào tài khoản

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH2_01	Chuyển tiền và lấy mã giao dịch MoMo để kiểm tra nạp tiền	Xuất hiện thông báo thành công với số tiền nạp, tiền được cộng vào	Nạp tiền thành công
TH2_02	Sử dụng một dãy số bất kỳ để kiểm tra nạp tiền	Hiển thị lỗi "Mã giao dịch không tồn tại"	Nạp tiền không thành công
TH2_03	Sử dụng một mã giao dịch bất kỳ trong MoMo mà không phải là chuyển cho số điện thoại như hướng dẫn	Hiển thị lỗi "Mã giao dịch không tồn tại"	Nạp tiền không thành công
TH2_04	Sử dụng lại một mã giao dịch đã sử dụng thành công ở TH2_01	Hiển thị lỗi "Mã giao dịch này đã được sử dụng"	Nạp tiền không thành công
TH2_05	Không nhập mã giao dịch	Hiển thị lỗi "TrainId Required"	Nạp tiền không thành công

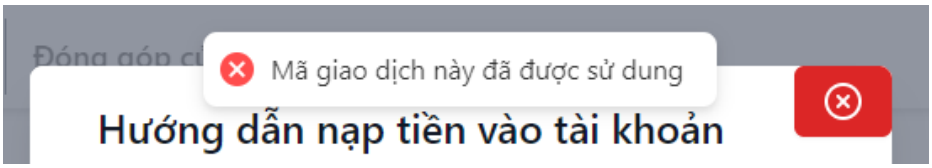
- Hình ảnh minh họa một số trường hợp

+ TH2_03



Hình 48: Minh họa giao dịch không tồn tại

+ TH2_04



Hình 49: Minh họa mã giao dịch đã được sử dụng

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng nạp tiền tự động hoạt động tốt.

4.8.3. Tạo lệnh rút tiền

- Mục tiêu: Kiểm thử chức năng rút tiền có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 14: Kiểm thử chức năng rút tiền

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH3_01	Tạo lệnh rút tiền với số tiền nhỏ hơn số tiền có trong tài khoản và đã thêm phương thức thanh toán	Xuất hiện thông báo "Tạo lệnh rút tiền thành công", số tiền bị trừ đúng với số tiền rút.	Rút tiền thành công
TH3_02	Tạo lệnh rút tiền với số tiền lớn hơn tài khoản và đã thêm phương thức thanh toán	Hiển thị lỗi "Số dư hiện tại không đủ"	Rút tiền không thành công
TH3_03	Tạo lệnh rút tiền với số tiền bỏ trống và đã thêm phương thức thanh toán	Hiển thị lỗi "Vui lòng nhập số tiền"	Rút tiền không thành công
TH3_04	Tạo lệnh rút tiền khi chưa thêm phương thức thanh toán	Hiển thị lỗi "Vui lòng thêm phương thức thanh toán để rút tiền"	Rút tiền không thành công

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng rút tiền hoạt động tốt.

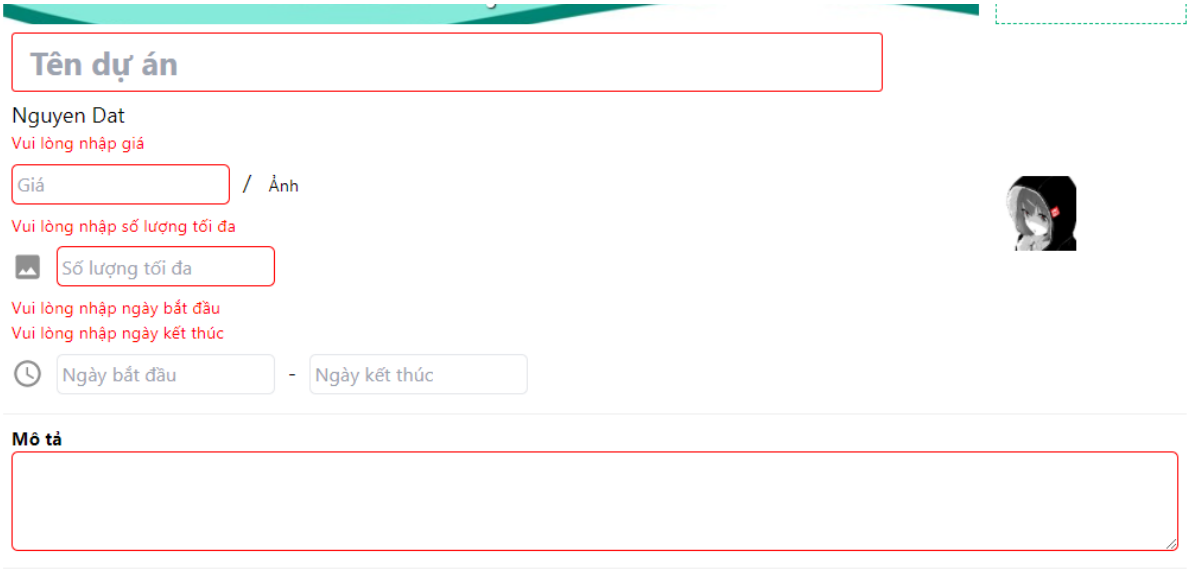
4.8.4. Tạo dự án mới

- Mục tiêu: Kiểm thử chức năng tạo dự án mới có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 15: Kiểm thử tạo dự án mới

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH4_01	Thực hiện tạo mới dự án với tất cả thông tin được điền đầy đủ	Xuất hiện thông báo tạo dự án thành công	Tạo dự án thành công
TH4_02	Thực hiện tạo mới dự án với tất cả thông tin bị bỏ trống	Hiển thị lỗi thiếu thông tin ở tất cả các trường.	Tạo dự án không thành công
TH4_03	Thực hiện tạo mới dự án với các kiểu dữ liệu không đúng (email)	Hiển thị lỗi sai kiểu dữ liệu	Tạo dự án không thành công

- Hình ảnh minh họa một số trường hợp: TH4_02



Hình 50: Minh họa lỗi thiếu thông tin

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng tạo dự án mới hoạt động tốt.

4.8.5. Gửi ảnh cho dự án

- Mục tiêu: Kiểm thử chức năng gửi ảnh cho dự án có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 16: Kiểm thử gửi ảnh cho dự án

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH5_01	Thực hiện gửi một và nhiều ảnh cho một dự án đang hoạt động.	Sau khoảng 10s đợi thì thông báo thành công xuất hiện	Gửi ảnh thành công
TH5_02	Thực hiện gửi ảnh cho một dự án tạm dừng hoạt động	Hiển thị thông báo lỗi "Dự án đã tạm dừng"	Gửi ảnh không thành công
TH5_03	Thực hiện gửi ảnh cho một dự án thu thập ảnh trên bản đồ trên nền tảng web	Hiển thị thông báo lỗi chỉ dành cho di động	Gửi ảnh không thành công
TH5_04	Thực hiện kéo một file không phải định dạng hình ảnh vào modal chọn ảnh	Hiển thị thông báo lỗi "Chỉ có thể tải lên: .jpg, .jpeg, .png, .gif"	Gửi ảnh không thành công
TH5_05	Gửi ảnh khi chưa đăng nhập tài khoản	Hiển thị lỗi chưa đăng nhập	Gửi ảnh không thành công

- Hình ảnh minh họa trường hợp: *TH5_04*

Hình 51: Minh họa lỗi sai định dạng

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng gửi ảnh hoạt động tốt.

4.8.6. Gửi ảnh trên bản đồ trên thiết bị di động

- Mục tiêu: Kiểm thử chức năng gửi ảnh trên bản đồ trên thiết bị di động cho dự án có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 17: Kiểm thử chức năng gửi hình trên di động

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH6_01	Thực hiện chụp ảnh tại vị trí hiện tại và chọn gửi ảnh.	Sau khoảng 10s đợi thì thông báo thành công xuất hiện	Gửi ảnh thành công
TH6_02	Thực hiện chụp ảnh sau khi di duyển khỏi vị trí ở TH6_01 và chọn gửi ảnh.	Sau khoảng 10s đợi thì thông báo thành công xuất hiện, đúng vị trí	Gửi ảnh thành công

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng gửi ảnh trên bản đồ trên thiết bị di động hoạt động tốt.

4.8.7. Thanh toán ảnh đã duyệt

- Mục tiêu: Kiểm thử chức năng thanh toán ảnh đã duyệt có hoạt động đúng hay không, các điều kiện đặt ra có hợp lệ hay không.

Bảng 18: Kiểm thử chức năng thanh toán

Trường hợp	Nội dung kiểm thử	Kết quả	Tình trạng
TH7_01	Thực hiện thanh toán cho một tài khoản *riêng lẻ* với số ảnh bất kỳ, số tiền trong khoản lớn hơn số phải trả	Hiển thị thông báo thành công, số tiền được trừ đúng với số phải trả, hình ảnh được bỏ watermark	Thanh toán thành công
TH7_02	Thực hiện thanh toán cho *tất cả* ảnh đã duyệt với số ảnh bất kỳ, số tiền trong khoản lớn hơn số phải trả	Hiển thị thông báo thành công, số tiền được trừ đúng với số phải trả, hình ảnh được bỏ watermark	Thanh toán thành công
TH7_03	Thực hiện thanh toán cho tất cả ảnh đã duyệt với số ảnh bất kỳ, số tiền trong khoản NHỎ hơn số phải trả	Hiển thị thông báo lỗi "Bạn không đủ tiền để thanh toán", ảnh vẫn giữ nguyên trạng thái	Thanh toán không thành công

- Kết luận: tất cả trường hợp kiểm thử đều hoạt động đúng với dự kiến, chức năng thanh toán ảnh đã duyệt hoạt động tốt.

PHẦN KẾT LUẬN

1. Kết luận

1.1. Kết quả đạt được

Qua quá trình thực hiện đề tài, em đã tích lũy cho bản thân nhiều kinh nghiệm và kiến thức trong các lĩnh vực như: sử dụng Prisma để thiết kế cơ sở dữ liệu, phân tích và thiết kế hệ thống, thiết kế API bằng NextJS Route, bổ sung kiến thức về ReactJS và React Native, sử dụng Tailwinds CSS để thiết kế giao diện, phân tích và giải quyết vấn đề, tự học và tìm kiếm tài liệu trên Internet.

Ngoài ra, còn giúp nâng cao kỹ năng trình bài chi tiết dự án của bản thân, cách tổ chức các thành phần của một dự án, cũng như phân chia thời gian cho từng thành phần.

1.2. Ưu điểm

- Kết quả là tạo được Web API services, giao diện web cho người dùng và admin, ứng dụng di động sử dụng API.

+ Web API services có chức năng liên kết trực tiếp với cơ sở dữ liệu. Thực hiện các phương thức xác thực nhằm đảm bảo không trả về thông tin nhạy cảm.

+ Giao diện Website: đơn giản, thân thiện với người dùng. Hiển thị tối đa những thông tin cần thiết, phân chia vai trò của mỗi loại người dùng.

+ Ứng dụng di động: nhằm tối đa lượng người dùng có thể tiếp cận được sản phẩm. Tạo tính thuận lợi khi mà đa số các ảnh hiện nay được chụp bởi điện thoại.

1.3. Hạn chế

Do không có nhiều thời gian để thiết kế và sự thiếu kinh nghiệm trong xây dựng website, dẫn đến một số hạn chế như:

- Chức năng phụ trợ vẫn còn ít, hiện tại chỉ hỗ trợ các chức năng căn bản để cho website hoạt động.

- Chưa thật sự tối ưu về tốc độ load trang cũng như thiếu độ mượt trong thao tác.

- Vẫn có thể xảy ra lỗi ở các trường hợp chưa được test kỹ.

- Ứng dụng cho Android vẫn chưa đầy đủ chức năng so với bản web.

- Ứng dụng cho IOS chưa được hoàn thành cho thiếu trang thiết bị cần thiết (máy Mac).

2. Hướng phát triển

- Giải quyết các hạn chế vừa nêu lên.

- Mở rộng nhiều loại thông tin có thể mua bán (văn bản, âm thanh,…).

- Thiết kế giao diện chuyên nghiệp hơn
- Tích hợp chức năng chat trực tiếp giữa người mua và người bán
- Thêm nhiều hình thức hỗ trợ đăng nhập (tài khoản, facebook,…)
- Hỗ trợ toàn bộ chức năng cho di động
- Quy định phạm vi thu thập hình ảnh trên bản đồ

TÀI LIỆU THAM KHẢO

[1] "ReactJS Documentation": https://reactjs.org/docs/

[2] "Prisma Documentation": https://www.prisma.io/

[3] "Uppy Documentation": https://uppy.io/docs

[4] "Next.js Documentation" https://nextjs.org/

[5] "Tailwindcss Documentation": https://tailwindcss.com/

[6] "Firebase Authentication Documentation": https://firebase.google.com/docs/auth

[7] "React Native Firebase": https://rnfirebase.io/

[8] "MoMo API": https://developers.momo.vn

[9] "Unofficial MoMo API": https://momofree.apimienphi.com/

[10] "Trove marketplace of Microsoft": https://www.microsoft.com/en-us/ai/trove

[11] "Server Side Rendering vs Client Side Rendering": https://viblo.asia/p/server-side-rendering-vs-client-side-rendering-RQqKLz30l7z