

ĐẠI HỌC BÁCH KHOA HÀ NỘI
Trường Công nghệ thông tin và Truyền thông



ĐỒ ÁN TỐT NGHIỆP

Đề tài: Website phân tích chứng khoán

NGUYỄN HỒNG SƠN
son.nh183976@sis.hust.edu.vn

**Ngành Công nghệ thông tin
Chuyên ngành Công nghệ công nghệ thông tin**

Giảng viên hướng dẫn: TS. Nguyễn Nhất Hải

Chữ ký GVHD

Khoa: Công nghệ phần mềm

Trường: Công nghệ thông tin và Truyền thông

HÀ NỘI, 08/2022

LỜI CAM KẾT

Họ và tên sinh viên:

Điện thoại liên lạc:

Email:

Lớp:

Hệ đào tạo:

Tôi – *Nguyễn Hồng Sơn* – cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *TS. Nguyễn Nhất Hải*. Các kết quả nêu trong ĐATN là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu, và các câu trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm với dù chỉ một sao chép vi phạm quy chế của nhà trường.

Hà Nội, ngày tháng năm

Tác giả ĐATN

Họ và tên sinh viên

LỜI CẢM ƠN

Em xin cảm ơn sự hướng dẫn của TS. Nguyễn Nhất Hải để hoàn thiện báo cáo đồ án này, cảm ơn các đồng nghiệp và anh chị tại công ty MDC đã hướng dẫn nội dung của đề tài.

Xin gửi lời cảm ơn tới gia đình và người thân đã luôn ủng hộ trong suốt những năm tháng đại học, cảm ơn những người bạn tốt mình đã may mắn gặp được tại Bách Khoa, và trên hết cảm ơn Đại học Bách Khoa đã cho em 4 năm sinh viên sóng gió và hạnh phúc xen lẫn.

TÓM TẮT NỘI DUNG ĐỒ ÁN

(i) Giới thiệu vấn đề

(a) Mục đích

Sản phẩm của đồ án này xây dựng là một website giúp người dùng phân tích chứng khoán trên thị trường, đồng thời theo dõi và hỗ trợ người dùng đưa ra quyết định đến hành động mua, bán của mình với các mã chứng khoán. Kèm theo đó website cũng cung cấp cho người dùng kiến thức để có thể hiểu và tiếp cận đúng vào thị trường chứng khoán tại Việt Nam thông qua video, hội thảo, bài viết, nghiên cứu ...

(b) Thực trạng

Đã có khá nhiều website phân tích chứng khoán trên thị trường và đã làm rất tốt, nhưng chưa có nhiều website đưa ra những bài học và hướng dẫn cho người dùng, nhất là đối tượng chưa có nhiều kinh nghiệm với thị trường chứng khoán. Cho nên đồ án lần này với mục tiêu đối tượng chính là cung cấp cho người dùng những kiến thức cơ bản lấp cao về thị trường chứng khoán.

(ii) Hướng tiếp cận

Xây dựng hệ thống website cho người dùng cuối tiếp cận đến những bài học, bài viết, khoá học và giúp họ theo dõi, phân tích số liệu tăng giảm của các mã chứng khoán trên thị trường cũng với những thông tin cần thiết.

(iii) Tổng quan giải pháp

(a) Về phía backend

Sử dụng Nodejs với framework express để xử lý các yêu cầu qua RESTful API. Cơ sở dữ liệu sử dụng MySQL. Lấy dữ liệu chứng khoán thời gian thực qua dữ liệu trả phí do Công ty SSI (Công ty cổ phần chứng khoán SSI) cung cấp

(b) Website cho người dùng cuối

Sử dụng framework Reactjs

(c) Website quản trị

Sử dụng framework Reactjs

(iv) Đóng góp chính

Xây dựng và hoàn thiện backend, lấy và lưu dữ liệu chứng khoán từ bên thứ ba.

Xây dựng website quản trị thông tin và quản lý người dùng. Xây dựng website

cung cấp thông tin đến người dùng cuối.

MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI.....	1
1.1 Đặt vấn đề.....	1
1.2 Mục tiêu và phạm vi đề tài.....	1
1.3 Định hướng giải pháp.....	1
1.4 Bố cục đồ án	1
CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU.....	3
2.1 Khảo sát hiện trạng	3
2.2 Tổng quan chức năng	3
2.2.1 Biểu đồ use case tổng quát	3
2.2.2 Biểu đồ use case phân rã Quản lý tài khoản	5
2.2.3 Biểu đồ use case phân rã Quản lý bài viết.....	6
2.2.4 Biểu đồ use case phân rã Quản lý danh sách các mã chứng khoán đầu tư.....	7
2.2.5 Biểu đồ use case phân rã Quản lý lịch sử giao dịch chứng khoán	7
2.2.6 Quy trình nghiệp vụ	8
2.3 Đặc tả chức năng	9
2.3.1 Đặc tả use case Tìm kiếm thông tin mã chứng khoán.....	9
2.3.2 Đặc tả use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán.....	10
2.3.3 Đặc tả use case Xác nhận thanh toán của người dùng.....	11
2.4 Yêu cầu phi chức năng	12
CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG.....	13
CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ	15
4.1 Thiết kế kiến trúc.....	15
4.1.1 Lựa chọn kiến trúc phần mềm	15

4.1.2 Thiết kế tổng quan.....	16
4.1.3 Thiết kế chi tiết gói	17
4.2 Thiết kế chi tiết.....	18
4.2.1 Thiết kế giao diện	18
4.2.2 Thiết kế lớp	22
4.2.3 Thiết kế cơ sở dữ liệu	26
4.3 Xây dựng ứng dụng.....	38
4.3.1 Thư viện và công cụ sử dụng	38
4.3.2 Kết quả đạt được	39
4.3.3 Minh họa các chức năng chính	39
4.4 Triển khai	41
CHƯƠNG 5. CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT	42
CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	45
6.1 Kết luận	45
6.2 Hướng phát triển.....	45
6.3 Lời kết	46

DANH MỤC HÌNH VẼ

Hình 2.1	Biểu đồ use case tổng quát cho người dùng	4
Hình 2.2	Biểu đồ use case tổng quát cho quản trị viên	5
Hình 2.3	Phân rã use case quản lý tài khoản	5
Hình 2.4	Phân rã use case quản lý bài viết	6
Hình 2.5	Phân rã use case quản lý danh sách các mã chứng khoán đầu tư	7
Hình 2.6	Phân rã use case quản lý lịch sử giao dịch chứng khoán	7
Hình 2.7	Quy trình Mua gói thành viên	8
Hình 2.8	Quy trình tạo và xem dữ liệu của danh mục	9
Hình 4.1	Mô hình Flux	15
Hình 4.2	Biểu độ phụ thuộc gói	16
Hình 4.3	Thiết kế chi tiết gói view	17
Hình 4.4	Thiết kế các gói store, view, api	18
Hình 4.5	Landing page	19
Hình 4.6	Màn hình tổng quan công ty	20
Hình 4.7	Màn hình chi tiết khoá học	20
Hình 4.8	Màn hình chi tiết bài viết	21
Hình 4.9	Màn hình phân tích danh mục	22
Hình 4.10	Sơ đồ lớp myListHomeScreen	23
Hình 4.11	Sơ đồ lớp StockInfoScreen	24
Hình 4.12	Biểu đồ giao tiếp của use case Xem thông tin mã chứng khoán	26
Hình 4.13	Biểu đồ giao tiếp của use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán	26
Hình 4.14	Sơ đồ thực thể liên kết - Liên quan đến cổ phiếu	27
Hình 4.15	Sơ đồ thực thể liên kết - Liên quan đến tài khoản	28
Hình 4.16	Sơ đồ cơ sở dữ liệu	29
Hình 4.17	Màn hình thông tin công ty	39
Hình 4.18	Phân tích danh mục	40
Hình 4.19	Phân tích lịch sử giao dịch	40
Hình 4.20	Chi tiết một bài viết	40
Hình 4.21	Chi tiết một khoá học	41
Hình 5.1	Tab cơ bản trong Danh mục của tôi	43

DANH MỤC BẢNG BIỂU

Bảng 2.1	Đặc tả use case Tìm kiếm thông tin mã chứng khoán	10
Bảng 2.2	Đặc tả use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán	11
Bảng 2.3	Đặc tả use case Xác nhận thanh toán của người dùng	12
Bảng 4.1	Thuộc tính của myListHomeScreen	23
Bảng 4.2	Phương thức của myListHomeScreen	24
Bảng 4.3	Thuộc tính của StockInfoScreen	25
Bảng 4.4	Phương thức của StockInfoScreen	25
Bảng 4.5	Danh sách thư viện và công cụ sử dụng	38
Bảng 4.6	Kết quả đạt được	39

DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

Viết tắt	Tên tiếng Anh	Tên tiếng Việt
API	Application Programming Interface	Giao diện lập trình ứng dụng
HTTP	Hyper Text Transfer Protocol	Giao thức truyền tải siêu văn bản
EUD	End-User Development	Phát triển ứng dụng người dùng cuối
GWT	Google Web Toolkit	Công cụ lập trình Javascript bằng Java của Google
HTML	HyperText Markup Language	Ngôn ngữ đánh dấu siêu văn bản
ORM	Object Relational Mapping	Một kĩ thuật lập trình, ánh xạ mỗi bản ghi trong cơ sở dữ liệu sang một đối tượng của lớp (class)
CNTT		Công nghệ thông tin
ĐATN		Đồ án tốt nghiệp
SV		Sinh viên
MG		MoneyGain
CSDL		Cơ sở dữ liệu
QTV		Quản trị viên

DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

Thuật ngữ	Ý nghĩa
Browser	Trình duyệt
Cache memory	Bộ nhớ đệm
Interpreter	Trình thông dịch
Compiler	Trình biên dịch
SQL	Ngôn ngữ truy vấn cơ sở dữ liệu
RESTful	Một quy chuẩn để thiết kế API
Backend	Hệ thống phục vụ các thông điệp yêu cầu thông qua một giao diện như API
Middleware	Những đoạn mã xử lý trung gian nằm giữa các yêu cầu (request) và phản hồi (response)
Frontend	Giao diện ứng dụng cho người dùng sử dụng (bao gồm cả quản trị và người dùng cuối)
Responsive	Phong cách thiết kế website trên trình duyệt của nhiều thiết bị với kích cỡ hiển thị khác nhau
Admin	Người dùng quản trị
System	Hệ thống (bao gồm cả backend và frontend)
Username	Tên đăng nhập của người dùng
Password	Mật khẩu đăng nhập của người dùng

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1 Đặt vấn đề

Thị trường chứng khoán từ lâu đã là mảnh đất làm tăng tài sản cá nhân cho các nhà đầu tư tài năng, cùng với sự cập nhật về thông tin nhanh chóng và nhiều như hiện nay, kèm theo đó là mức thu nhập của người dân đã tăng trong thời gian gần đây, thì nhu cầu tiếp cận và tham gia vào thị trường chứng khoán đã mở rộng ra từ những doanh nhân, những nhà đầu tư có thẩm niêm đến một bộ phận lớn cá nhân, nhà đầu tư trẻ muốn thử sức và tìm hiểu cơ hội kiếm thêm thu nhập từ thị trường này.

Mặc dù vậy, để tham gia vào môi trường khắc nghiệt như chứng khoán không phải là một điều dễ dàng, và có được lợi nhuận từ nó càng rất khó khăn ngay cả với những người có kinh nghiệm. Một nhu cầu cấp thiết đặt ra là giúp những người có nhu cầu tham gia thị trường chứng khoán nhưng chưa có nhiều kinh nghiệm có được những thông tin cần thiết làm hành trang cho sự đầu tư của mình được đặt ra.

1.2 Mục tiêu và phạm vi đề tài

Về việc giúp người dùng theo dõi sự tăng giảm điểm cũng như biến động của chứng khoán đồng thời đưa ra cho họ kiến thức cần thiết, website vietstockfinance.com.vn của doanh nghiệp Vietstock đã làm rất tốt yêu cầu được đề ra.

Đồ án ứng dụng lần này nhằm tạo ra một sản phẩm cạnh tranh với website của Vietstock từ chính yêu cầu của khách hàng thực tế (doanh nghiệp MoneyGain). Sản phẩm sẽ tối ưu lượng thông tin để gửi đến cho người dùng cuối, tóm lược những thông tin cần thiết nhất giúp người dùng tham gia vào thị trường với một giao diện thân thiện và dễ hiểu.

1.3 Định hướng giải pháp

Reactjs là một công nghệ để xây dựng hệ thống website frontend nhanh chóng, dễ dàng và dễ học. Về phía backend em chọn Nodejs để xây dựng, cơ sở dữ liệu (CSDL) là MySQL.

Ưu điểm của Reactjs cũng như Nodejs chính là sự dễ tiếp cận và học hỏi cũng như xây dựng hệ thống mà nó mang lại, với một cộng đồng phát triển lớn.

Kết quả đạt được của đồ án là một hệ thống hỗ trợ cùng website quản trị và website cho người dùng cuối trong việc phân tích chứng khoán.

1.4 Bố cục đồ án

Phần còn lại của báo cáo đồ án tốt nghiệp này được tổ chức như sau.

Chương 2 trình bày về phân tích thiết kế các use cases, luồng nghiệp vụ chính có trong hệ thống.

Chương 3 sẽ trình bày về những nền tảng công nghệ được sử dụng và ứng dụng của chúng để giải quyết vấn đề trong đồ án.

Chương 4 trình bày về phân tích kiến trúc phần mềm của hệ thống và thiết kế chi tiết của nó.

Chương 5 trình bày về những vướng mắc gặp phải trong quá trình thực hiện đồ án và cách giải quyết.

Chương 6 là kết luận về kết quả, đóng góp cũng như những điều mà đồ án đã làm tốt và những điều cần cải thiện.

CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

2.1 Khảo sát hiện trạng

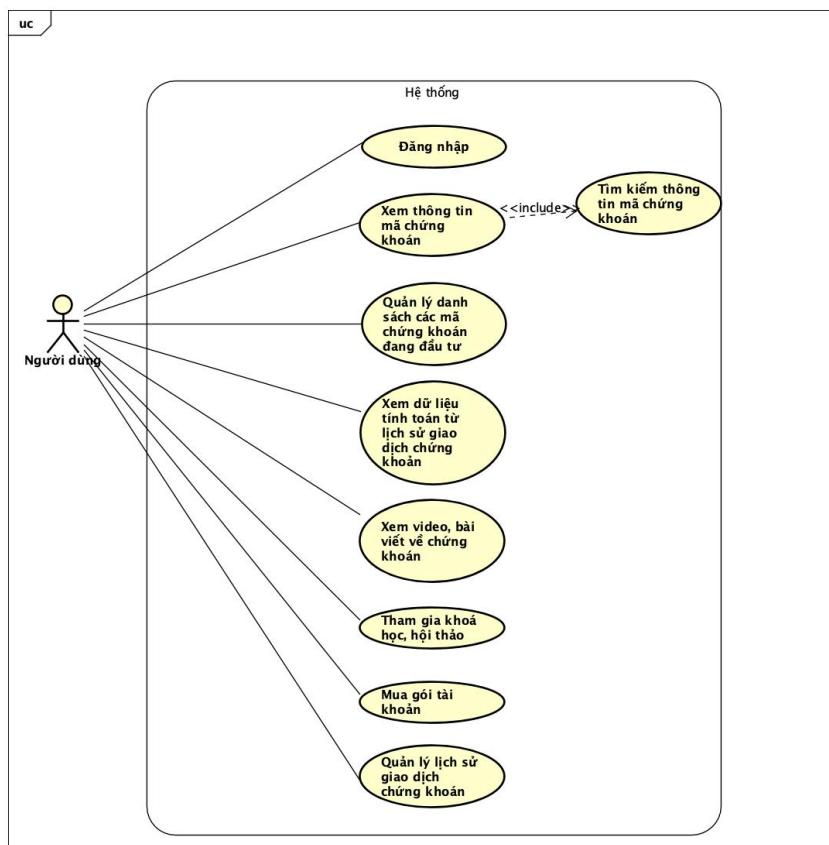
Sản phẩm kết quả của đồ án được tham khảo trực tiếp từ các chức năng của website vietstockfinance.com.vn và được thay đổi một số chức năng cũng như tinh chỉnh các chức năng quan trọng mà một website phân tích tài chính cần có.

- Về chức năng của website cho người dùng cuối:
 - (i) Đăng nhập, đăng ký tài khoản
 - (ii) Mua gói tài khoản
 - (iii) Theo dõi lịch sử, biến động của các mã cổ phiếu, chứng khoán
 - (iv) Thông kê tài chính của công ty phát hành cổ phiếu
 - (v) Cung cấp cho người dùng kiến thức, thông tin liên quan đến chứng khoán
 - (vi) Tính toán cho người dùng dữ liệu dựa trên số liệu về những giao dịch mua bán của người dùng (dữ liệu do người dùng cung cấp)
- Về chức năng của website quản trị:
 - (i) Quản lý tài khoản
 - (ii) Quản lý các thông tin, sản phẩm cung cấp đến cho người dùng

2.2 Tổng quan chức năng

2.2.1 Biểu đồ use case tổng quát

- Tác nhân là người dùng

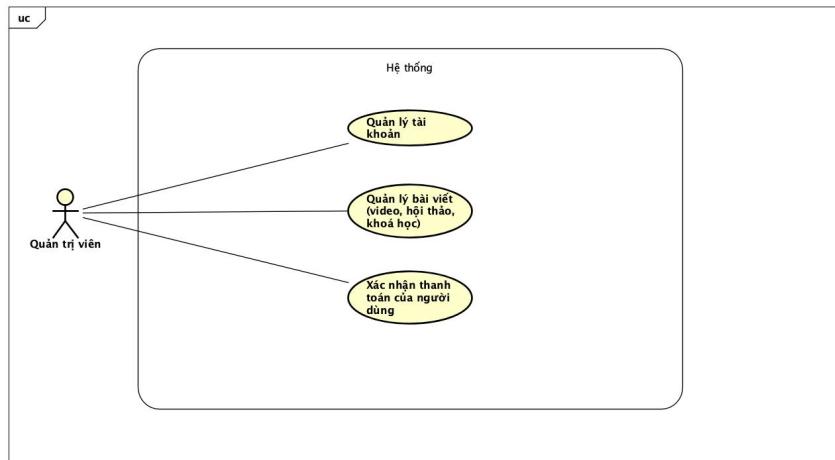
**Hình 2.1:** Biểu đồ use case tổng quát cho người dùng

Mô tả:

- Đăng nhập: Người dùng đăng nhập bằng email và mật khẩu của tài khoản
- Xem thông tin một mã chứng khoán
- Tìm kiếm thông tin mã chứng khoán: Người dùng tìm kiếm một mã chứng khoán
- Quản lý danh sách các mã chứng khoán đang đầu tư: Người dùng thực hiện thêm, sửa, xoá danh sách lưu các lịch sử giao dịch chứng khoán của mình
- Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán: Hệ thống tính toán số liệu dựa trên dữ liệu người dùng cung cấp, từ đó đưa ra nhận định tổng quan về quá trình giao dịch của họ
- Xem video, bài viết về chứng khoán: Người dùng xem video, bài viết về chứng khoán trên giao diện website
- Tham gia hội thảo, khoá học: Người dùng đăng ký tham gia hội thảo, khoá học (sẽ phải được xác nhận thanh toán nếu hội thảo, khoá học cần trả phí)
- Mua gói tài khoản: Người dùng mua (gia hạn nếu đã mua trước đó) gói tài

khoản (thành công khi quản trị viên xác nhận thanh toán)

- Quản lý lịch sử giao dịch chứng khoán: Người dùng thêm hoặc sửa, xoá thông tin giao dịch đã cung cấp trước đó

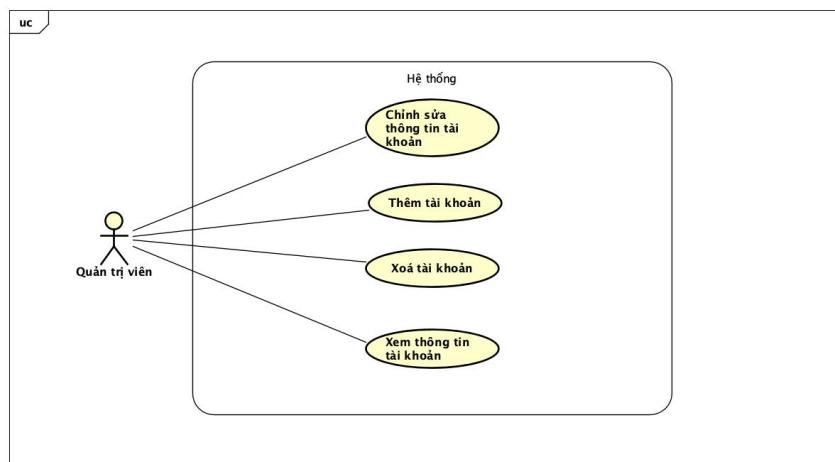


Hình 2.2: Biểu đồ use case tổng quát cho quản trị viên

Mô tả:

- Quản lý tài khoản
 - Quản lý video, bài viết, hội thảo, khoá học
 - Xác nhận thanh toán của người dùng: Quản trị viên xác nhận thanh toán của người dùng (thành công, thất bại)

2.2.2 Biểu đồ use case phân rã Quản lý tài khoản



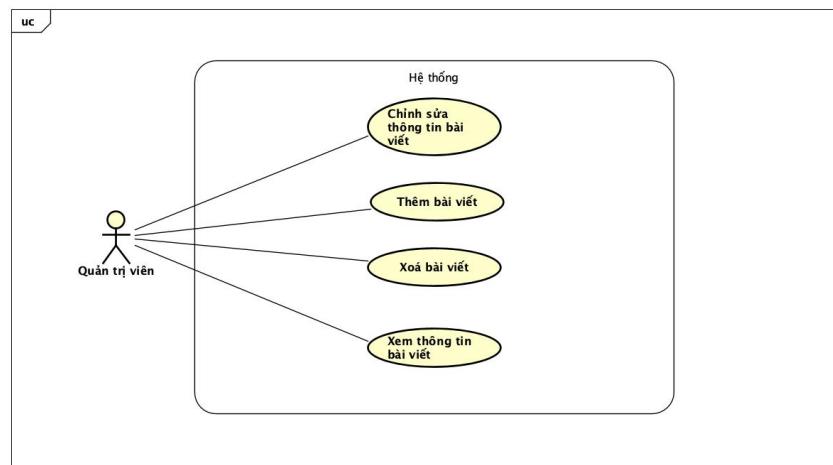
Hình 2.3: Phân rã use case quản lý tài khoản

Mô tả

- Chỉnh sửa thông tin tài khoản
- Thêm tài khoản
- Xoá tài khoản
- Xem thông tin tài khoản

Các use case quản trị viên tham gia trên đây áp dụng cho mọi loại tài khoản.

2.2.3 Biểu đồ use case phân rã Quản lý bài viết

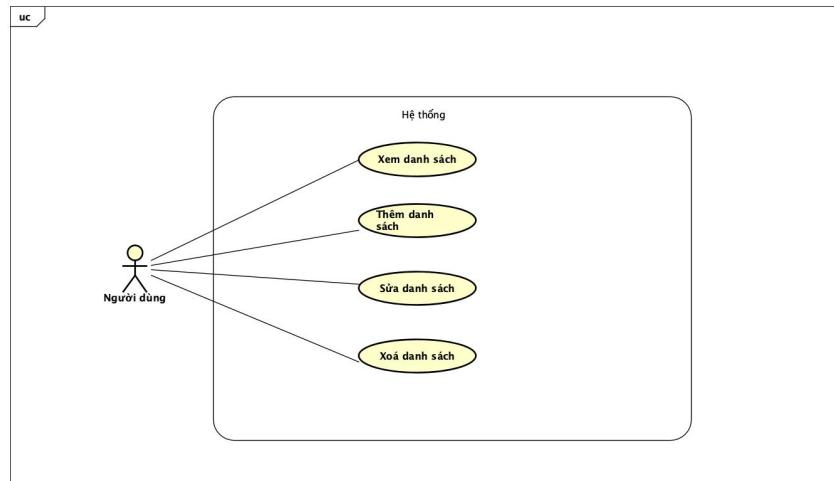


Hình 2.4: Phân rã use case quản lý bài viết

Mô tả

- Chỉnh sửa thông tin bài viết
- Thêm bài viết
- Xoá bài viết
- Xem thông tin bài viết

2.2.4 Biểu đồ use case phân rã Quản lý danh sách các mã chứng khoán đầu tư

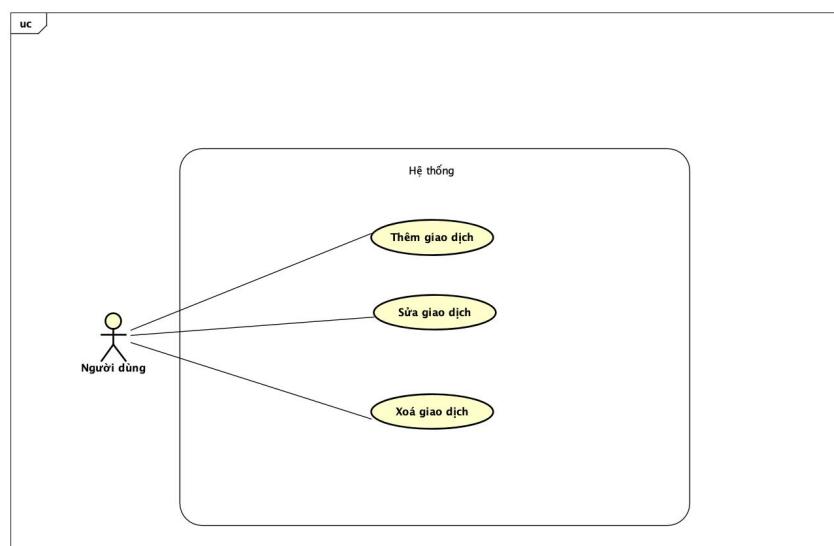


Hình 2.5: Phân rã use case quản lý danh sách các mã chứng khoán đầu tư

Mô tả

- Chỉnh sửa thông tin danh sách
- Thêm danh sách
- Xoá danh sách
- Xem thông tin danh sách

2.2.5 Biểu đồ use case phân rã Quản lý lịch sử giao dịch chứng khoán



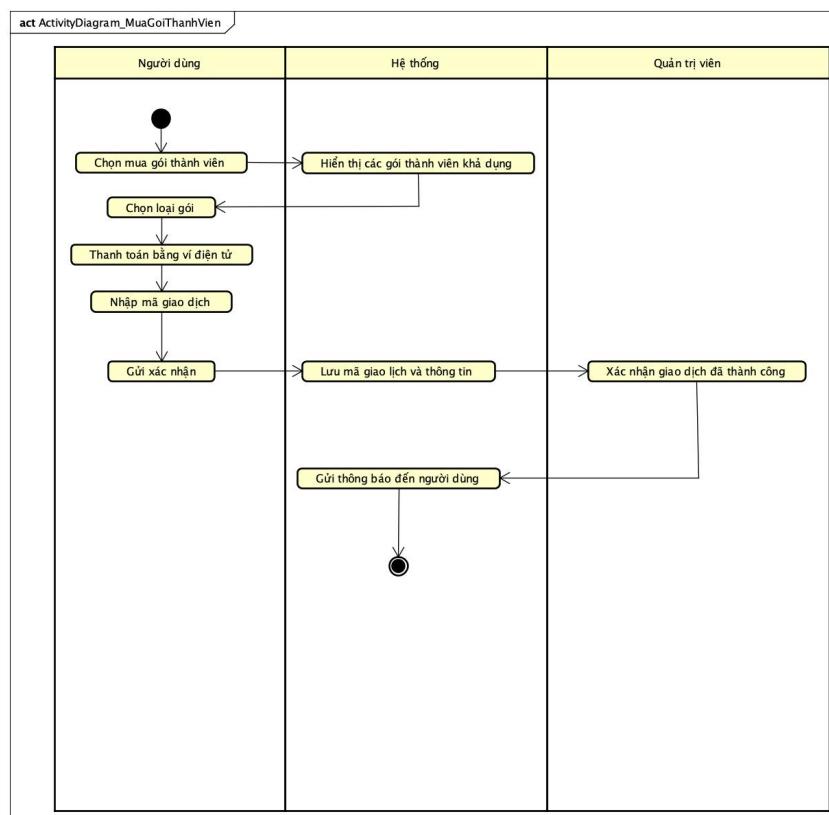
Hình 2.6: Phân rã use case quản lý lịch sử giao dịch chứng khoán

Mô tả

- Chỉnh sửa thông tin giao dịch
- Thêm giao dịch
- Xoá giao dịch

2.2.6 Quy trình nghiệp vụ

1. Quy trình nghiệp vụ Mua gói thành viên

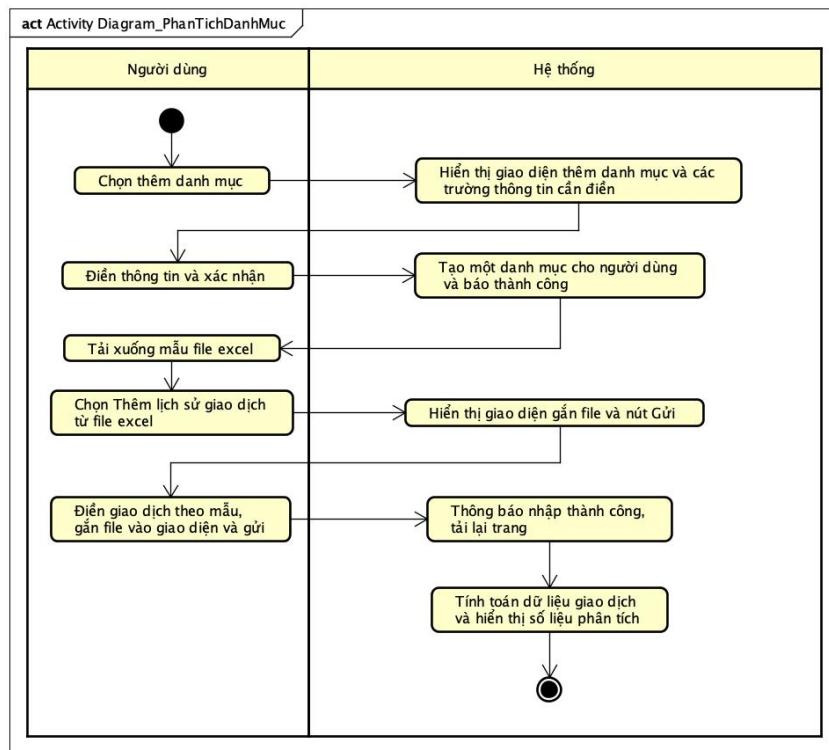


Hình 2.7: Quy trình Mua gói thành viên

Mô tả:

- Khi người dùng ấn nâng cấp (hoặc gia hạn) tài khoản, hệ thống sẽ hiển thị các gói tài khoản và thông tin từng gói.
- Người dùng chọn một gói sẽ được dẫn đến trang để nhập mã giao dịch mua thành công. Thông tin các ví điện tử để nhận chuyển khoản sẽ được cung cấp và người dùng có thể chọn một trong các tài khoản.
- Sau khi chuyển khoản thành công thì người dùng nhập mã giao dịch đó và gửi lên hệ thống, hệ thống sẽ lưu lại và đợi xác nhận từ phía quản trị viên. Sau khi quản trị viên xác nhận sẽ có thông báo đến người dùng.

2. Quy trình nghiệp vụ Tạo và xem thông tin tính toán từ danh mục lịch sử giao dịch



Hình 2.8: Quy trình tạo và xem dữ liệu của danh mục

Mô tả:

- Người dùng sẽ thêm một danh mục mới (nếu chưa có) và thêm lịch sử giao dịch bằng cách nhập vào file excel và gửi lên hệ thống.
- Hệ thống tạo mới danh mục và các dữ liệu giao dịch từ người dùng cung cấp.
- Hệ thống tính toán số liệu từ các giao dịch và hiển thị thông tin cho người dùng.

2.3 Đặc tả chức năng

2.3.1 Đặc tả use case Tìm kiếm thông tin mã chứng khoán

Tên use case	Tìm kiếm thông tin một mã chứng khoán		Mã use case	uc01
Tác nhân	Người dùng			
Tiền điều kiện	Không			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1	Người dùng	Ấn nút tìm kiếm	
	2	Hệ thống	Hiện thanh tìm kiếm	
	3	Người dùng	Nhập từ khoá	
	4	Hệ thống	Hiển thị danh sách thoả mãn từ khoá	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	4 . 1	Hệ thống	Hiển thị "Không có kết quả phù hợp"	
Hậu điều kiện	Không			

Bảng 2.1: ĐẶC TẢ USE CASE TÌM KIẾM THÔNG TIN MÃ CHỨNG KHOÁN

2.3.2 ĐẶC TẢ USE CASE XEM DỮ LIỆU TÍNH TOÁN TỪ LỊCH SỬ GIAO DỊCH CHỨNG KHOÁN

Tên use case	Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán		Mã use case	uc02
Tác nhân	Người dùng			
Tiền điều kiện	Người dùng có ít nhất một danh sách có ít nhất một dữ liệu giao dịch với mã chứng khoán hợp lệ bất kì			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1	Người dùng	Ấn nút "Danh mục của tôi"	
	2	Hệ thống	Hiện màn hình danh mục của tôi	
	3	Người dùng	Chọn từng loại tab thống kê dữ liệu	
	4	Hệ thống	Hiển thị dữ liệu tính toán dựa trên lịch sử giao dịch và thông tin về các mã chứng khoán	
Luồng sự kiện thay thế	Không có			
Hậu điều kiện	Không			

Bảng 2.2: Đặc tả use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán

2.3.3 Đặc tả use case Xác nhận thanh toán của người dùng

Tên use case	Xác nhận thanh toán của người dùng		Mã use case	uc03
Tác nhân	Quản trị viên			
Tiền điều kiện	Có ít nhất một bản ghi có mã giao dịch với tài khoản người dùng hợp lệ			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1	Quản trị viên	Chọn xem danh sách giao dịch	
	2	Quản trị viên	Chọn xác nhận giao dịch cần xác nhận với trạng thái "Thành Công"	
	3	Hệ thống	Hỏi xác nhận	
	4	Quản trị viên	Xác nhận đồng ý	
	5	Hệ thống	Hiển thị thông báo thành công	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	5.1	Hệ thống	Hiển thị thông báo xác nhận thất bại	
Hậu điều kiện	Không			

Bảng 2.3: Đặc tả use case Xác nhận thanh toán của người dùng

2.4 Yêu cầu phi chức năng

- Phía frontend:
 - Giao diện có bố cục hợp lý, dễ quan sát.
 - Giao diện phải ổn định với các trình duyệt web phổ biến (Chrome, Safari, Firefox, Edge)
- Phía backend:
 - Hiệu suất truy cập cơ sở dữ liệu đáp ứng tối thiểu 1000 truy vấn đồng thời.
 - Server không được có thời gian chết hoặc tối đa là 2 giây.
 - Phải quản lý được mọi dữ liệu do hệ thống tạo ra (như tài khoản, bài viết) trên giao diện trang quản lý cho quản trị viên.
 - Đảm bảo an toàn bảo mật, tránh truy cập trái phép hay làm hỏng dữ liệu từ mã độc (như SQL Injection).

CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG

1. Phía backend:

- (a) **Nodejs** là một nền tảng được xây dựng trên “V8 Javascript engine” được viết bằng **C++** và **Javascript**. Sự dễ học, tiện lợi, thời gian phát triển nhanh cùng với một cộng đồng phát triển lớn là lý do đây là lựa chọn để xây dựng backend.

Tốc độ tính toán nhanh là ưu điểm của **Nodejs**, ngay cả khi thực hiện tính toán dữ liệu giao dịch của người dùng kết hợp với lấy số liệu thực tế từ các mã chứng khoán vẫn cho ra phản hồi rất nhanh chóng.

- (b) **Express** là một framework được xây dựng trên nền tảng của **Nodejs**. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile.

Expressjs hỗ trợ các method **HTTP** và **middleware** tạo ra **API** vô cùng mạnh mẽ và dễ sử dụng.

Phía backend sử dụng framework này để xử lý các yêu cầu HTTP từ phía client.

- (c) **Sequelize** là một ORM dành cho **Node.js** giúp truy cập nhanh chóng đến nhiều loại cơ sở dữ liệu khác nhau.

Server kết nối và truy vấn với cơ sở dữ liệu qua **Sequelize**, các bản ghi khi truy vấn sẽ là một đối tượng được định nghĩa sẵn khi khởi tạo.

- (d) **MySQL** là một trong những hệ thống quản lý cơ sở dữ liệu (CSDL) mã nguồn mở quan hệ SQL sử dụng trên web phổ biến nhất hiện nay.

Indexing là cách sắp xếp các bản ghi trên nhiều trường. Tạo **index** trên một trường của một bảng là việc tạo ra một cấu trúc dữ liệu khác giữ giá trị của trường, và trả tới bản ghi mà nó liên quan tới. **Index** đó sẽ được sắp xếp, và cho phép tìm kiếm nhị phân thực hiện. Nhược điểm của các **index** này là yêu cầu thêm không gian bộ nhớ, vì các index được lưu trữ cùng nhau trên một bảng sử dụng MyISAM engine, file này có thể nhanh chóng đạt đến kích thước giới hạn của hệ thống, nếu có quá nhiều trường của cùng một table được đánh **index**.

Việc đánh **index** cho các bảng chứa số lượng bản ghi lớn như dữ liệu

biến động của các mã chứng khoản (khoảng 300 triệu bản ghi) giúp cho thời gian truy vấn cải thiện rất rõ rệt. Tuy nhiên hiệu quả chỉ với những truy vấn không quá phức tạp, vẫn nên tránh những truy vấn có độ phức tạp cao và phải liên kết nhiều bảng dữ liệu khi thực hiện truy vấn.

2. Phía frontend:

- (a) **React.js** là một thư viện **Javascript** đang nổi lên trong những năm gần đây với xu hướng **Single Page Application**. Trong khi những **framework** khác cố gắng hướng đến một mô hình **MVC** hoàn thiện thì **React** nổi bật với sự đơn giản và dễ dàng phối hợp với những thư viện **Javascript** khác.

React là một thư viện UI phát triển tại **Facebook** để hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được.

Ưu điểm khi sử dụng **React** là tốc độ phát triển, nhiều thư viện sẵn có, yêu cầu nắm vững kiến thức cơ bản về **CSS, Javascript, HTML**.

- (b) **Redux** là một thư viện **Javascript** giúp tạo ra thành một lớp quản lý trạng thái của ứng dụng.

Redux được xây dựng dựa trên 3 nguyên lý:

- Nguồn dữ liệu tin cậy duy nhất: **State** - hay trạng thái, của toàn bộ ứng dụng được chứa trong một object tree, cây đối tượng, nằm trong **Store** (lưu trữ) duy nhất.
- Trạng thái chỉ được phép đọc: Cách duy nhất để thay đổi State của ứng dụng là phát một Action (là 1 đối tượng mô tả những gì xảy ra).
- Thay đổi chỉ bằng hàm thuần túy: Để chỉ ra cách mà State được biến đổi bởi Action chúng ta dùng các pure function (cách hàm biến đổi thông thường) gọi là **Reducer**.

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

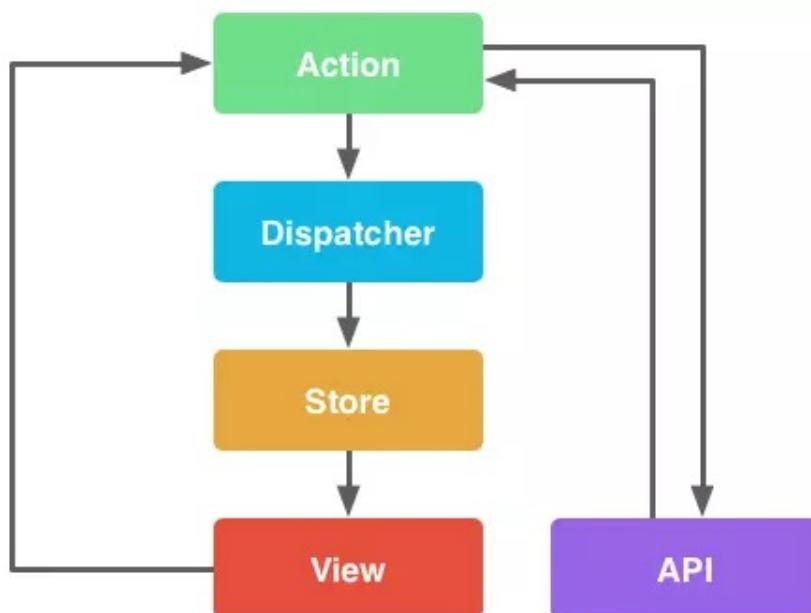
4.1 Thiết kế kiến trúc

4.1.1 Lựa chọn kiến trúc phần mềm

Flux là một kiến trúc mà **Facebook** sử dụng trong khi làm việc với **React**, là một kiểu kiến trúc mới hỗ trợ thêm cho **React**, đồng thời xây dựng ý tưởng về luồng dữ liệu một chiều (Unidirectional Data Flow).

Một kiến trúc Flux điển hình là sự kết hợp giữa thư viện **Dispatcher** (được viết bởi **Facebook**) cùng với module **NodeJS EventEmitter** để tạo nên một hệ thống sự kiện (Event System) quản lý các trạng thái của ứng dụng.

Mô hình hoạt động:



Hình 4.1: Mô hình Flux

Ta có thể hiểu đơn giản như sau :

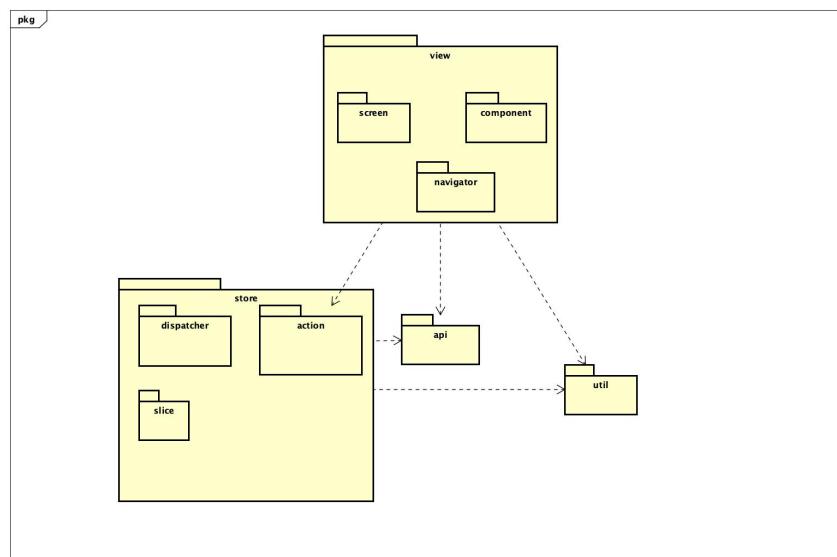
- **Views** chính là thành phần làm nhiệm vụ hiển thị nội dung ứng dụng (có thể hiểu giống như thành phần V trong mô hình MVC).
- Khi người dùng tương tác với ứng dụng làm thay đổi trạng thái (state) của ứng dụng (VD: thêm, sửa, xóa dữ liệu cá nhân), View sẽ thông qua Action (hành động) gửi các thông tin thay đổi tới **Dispatcher** gồm có :
 - **action_name**: tên của Action (VD: **ADD_ITEM** - thêm sản phẩm vào giỏ hàng).

- **action_payload**: thông tin chi tiết nội dung muốn gửi (VD: Đối tượng chứa thông tin ID, quantity, price, ... của sản phẩm).
- Sau khi nhận được thông tin từ **Action, Dispatcher** làm nhiệm vụ truyền tải (broadcast) nội dung nhận được tới các **Store** đăng ký lắng nghe sự kiện thay đổi từ trước đó.
- Store sau khi nhận thông tin, tiến hành cập nhật dữ liệu (có thể hiểu việc cập nhật dữ liệu ở đây giống việc cập nhật state của Component).
- Sau khi cập nhật, Store bắn sự kiện xuống **View** để tiến hành cập nhật hiển thị cho người dùng.

Ngoài ra trong sơ đồ trên còn có một thành phần API để lấy dữ liệu từ server. Sơ đồ trên đảm bảo luồng dữ liệu di chuyển trong Flux bắt buộc đi theo một đường nhất định.

Đồ án này cũng đã đi theo kiến trúc Flux thiết kế cho React. Các View, Store tổng thể, các Dispatcher và Action gọi đến các API được thiết kế và quan hệ rõ ràng.

4.1.2 Thiết kế tổng quan



Hình 4.2: Biểu đồ phụ thuộc gói

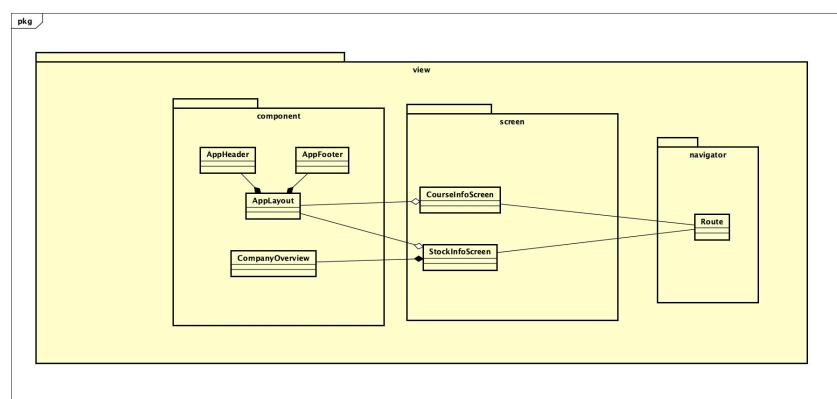
Mô tả:

- Gói view: chứa hai gói navigator, screen và component. Gói component gồm những component viết riêng cho việc tái sử dụng. Gói screen gồm các màn hình giao diện và gói navigator chứa trật tự, đường dẫn đến các màn hình trong screen.

- Gói util chứa các hàm tính toán giúp cho việc lập trình thuận tiện đồng thời giúp mã nguồn gọn gàng.
- Gói store chứa các gói action, dispatcher và các slice (cấu thành nên Store của toàn bộ ứng dụng). Các đối tượng trong component, hay screen sẽ sử dụng các đối tượng trong action để thay đổi giao diện, cập nhật trạng thái.
- Gói api chứa các phương thức giao tiếp api trực tiếp với server. View có thể lấy dữ liệu trực tiếp với các đối tượng trong api, hoặc có thể thông qua action để cập nhật trạng thái.

4.1.3 Thiết kế chi tiết gói

- Thiết kế gói view



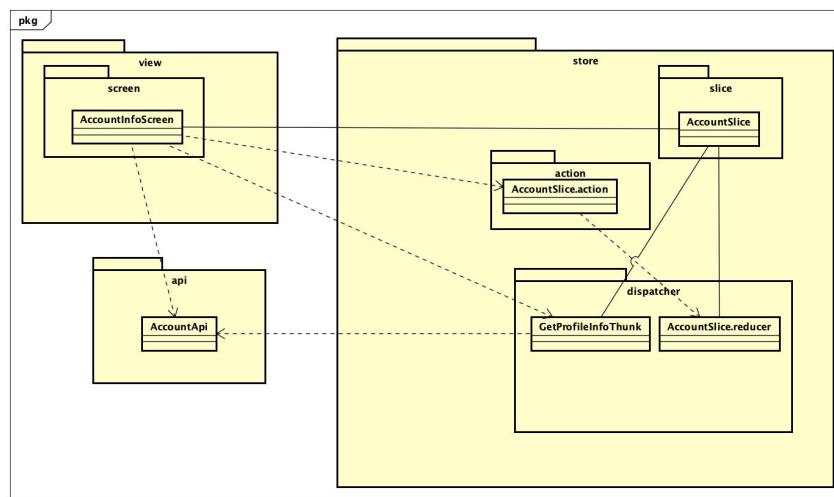
Hình 4.3: Thiết kế chi tiết gói view

Mô tả:

- Các lớp `AppHeader`, `AppFooter` hợp thành nên `AppLayout`, dùng cho mọi màn hình trong ứng dụng.
- Hai lớp là `CourseInfoScreen` và `StockInfoScreen` đều tạo nên từ `AppLayout`, và `StockInfoScreen` được cấu thành từ `CompanyOverview`.
- Các lớp trong screen, như `CourseInfoScreen` và `StockInfoScreen` đều liên kết với `Route` trong navigator, thực hiện việc điều hướng qua các màn hình.

Trên đây là mô tả liên kết giữa các lớp trong ba gói nhỏ trong gói view bằng minh họa với một vài lớp đặc trưng cho từng gói.

- Thiết kế gói store, view, api

**Hình 4.4:** Thiết kế các gói store, view, api

Mô tả:

- AccountInfoScreen có thể dùng AccountSliceAction trong gói action hoặc GetProfileThunk để thay đổi trạng thái của AccountSlice, hoặc có thể chỉ sử dụng AccountApi để lấy về thông tin mà không thay đổi trạng thái.
- AccountInfoScreen có thể lắng nghe thay đổi trạng thái của AccountSlice xử lý giao diện cho phù hợp.
- Việc các lớp trong view tác động tới store thông qua dispatcher (hay action) để cập nhật trạng thái vào store, rồi thông qua store để thay đổi giao diện như vậy là nguyên tắc chung của mô hình Flux (như giới thiệu trong mục 4.1.1)

4.2 Thiết kế chi tiết

4.2.1 Thiết kế giao diện

Chuẩn hóa màn hình

- **Hiển thị**

Số lượng màu được hỗ trợ: 16,777,216 màu

Bố cục: responsive cho màn hình desktop và thiết bị di động

- **Màn hình**

Vị trí của cửa sổ hội thoại (dialog): chính giữa màn hình

Vị trí của thông báo: ở góc phía trên bên phải màn hình

Thông nhất thiết kế

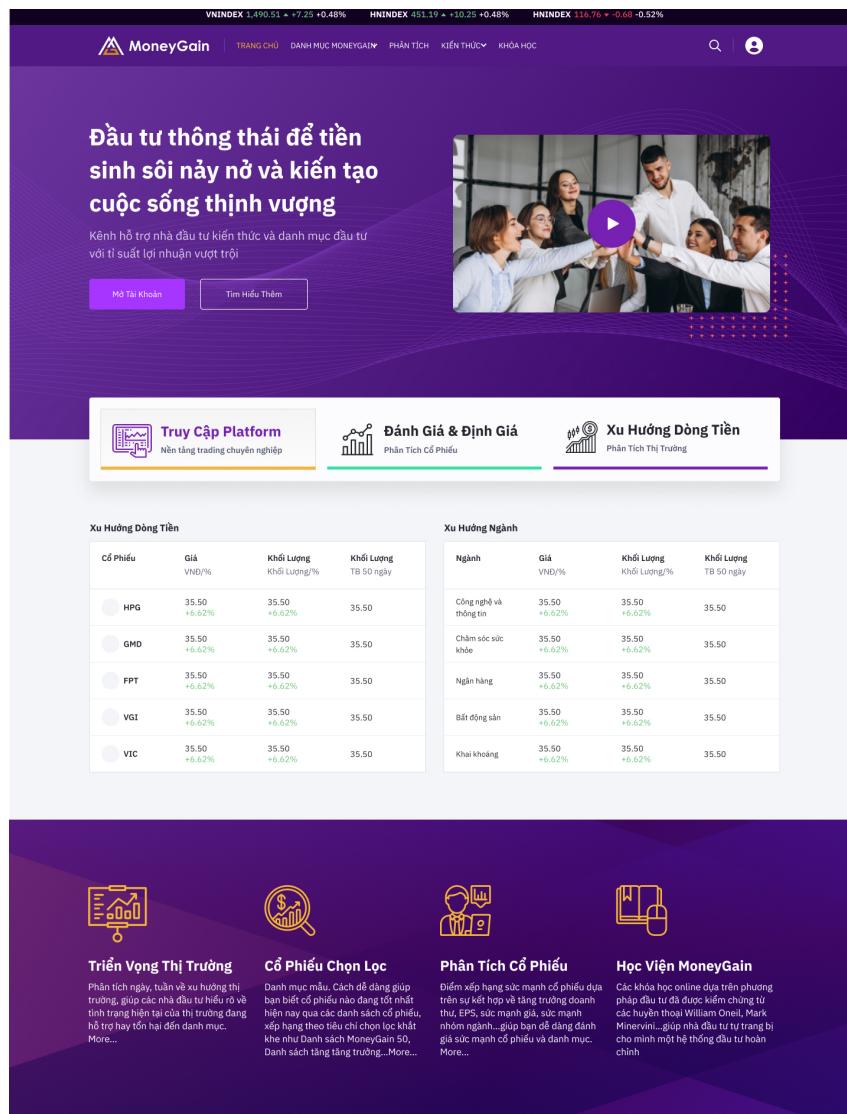
- **Màu sắc chủ đạo:** #7621B1 hay rgba(118, 33, 177, 1)

- Font chữ: IBM Plex Sans

Thiết kế giao diện

Công cụ: Figma

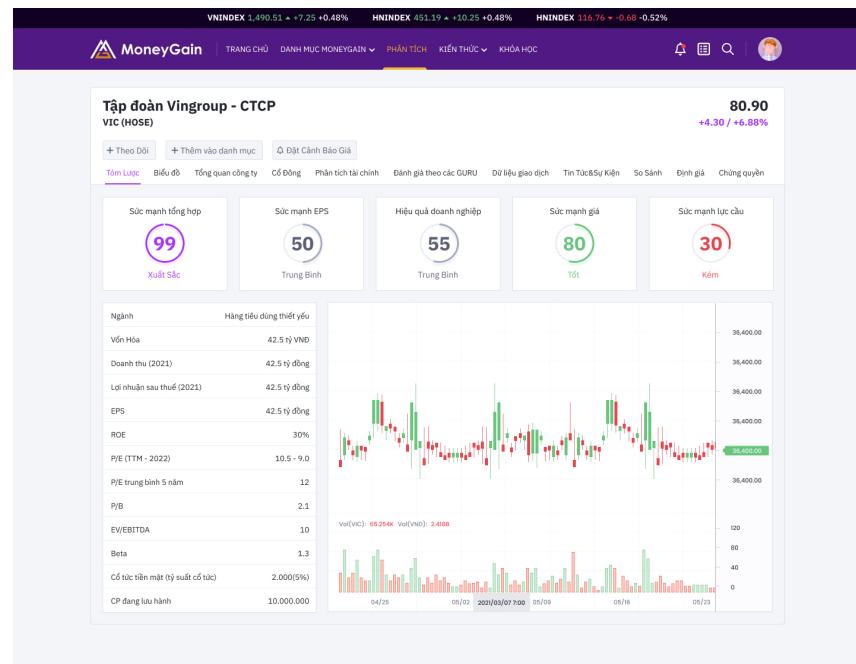
- Landing page



Hình 4.5: Landing page

Mô tả: Thông tin của các mã chứng khoán nổi bật sẽ chạy theo chiều ngang phía trên màn hình

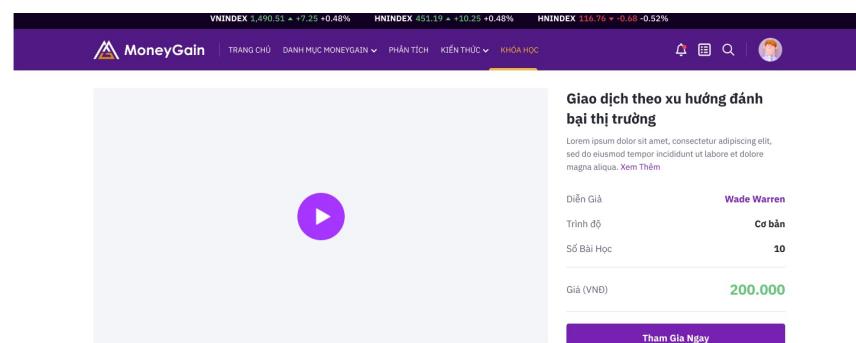
- Tổng quan công ty



Hình 4.6: Màn hình tổng quan công ty

Mô tả: Hiện thông tin cơ bản và thay đổi của mã cổ phiếu của công ty

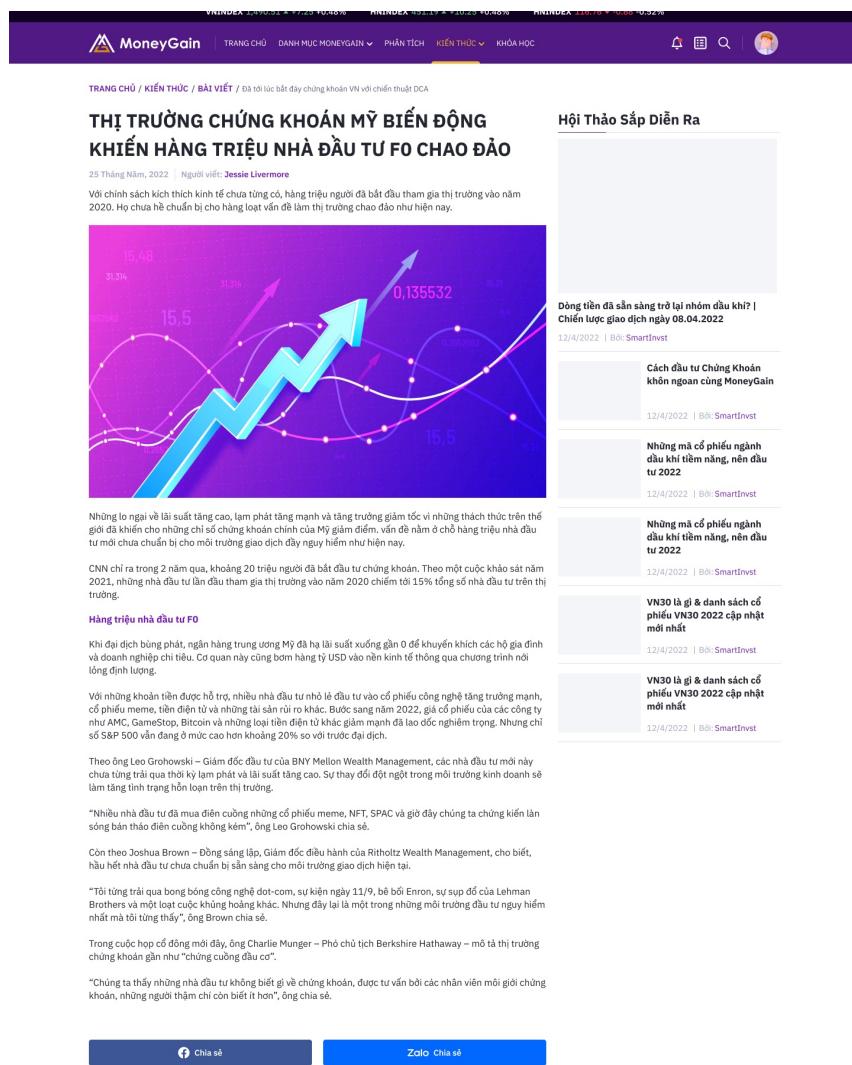
- **Chi tiết khoá học**



Hình 4.7: Màn hình chi tiết khoá học

Mô tả: Thông tin của khoá học và video giới thiệu

- **Chi tiết bài viết**



Hình 4.8: Màn hình chi tiết bài viết

• **Danh mục**

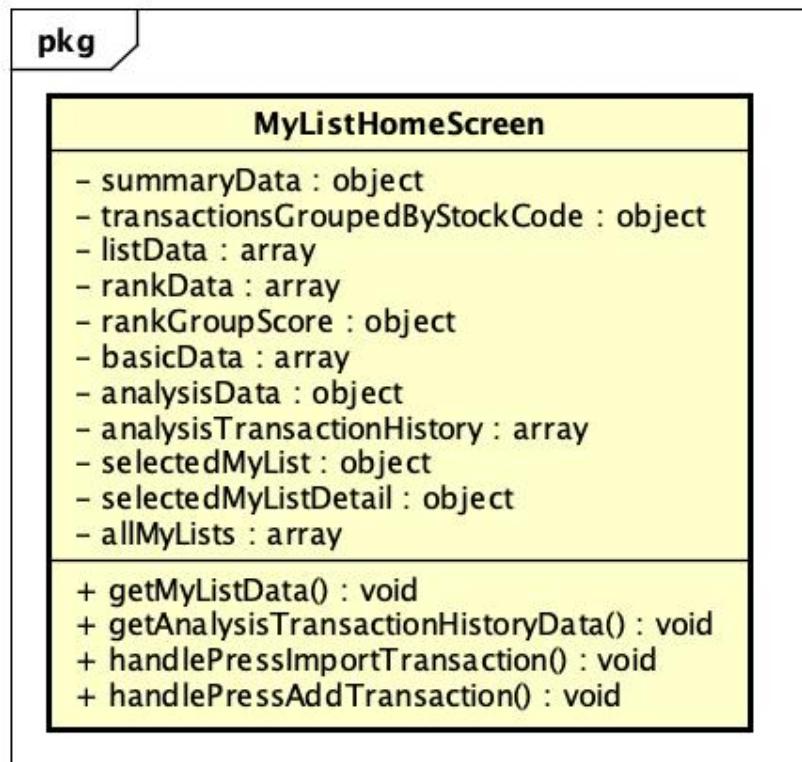
Mã Tên Công Ty	Tổng Khối Lượng Cổ Phiếu	Giá TB (VND) Mua/Bán	Giá Trị (VND) Mua/Bán	Giá Thị Trường VND	Lãi/Lỗ %/VND	Lãi/Lỗ Trong Ngày %/VND	Thao Tác
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***
VCB CÔNG TY CỔ PHẦN TẬP...	50.000	58.55	58.90	60.00	+2.44% +950	+2.44% +950	***

Hình 4.9: Màn hình phân tích danh mục

Mô tả: Phân tích từ dữ liệu do người dùng cung cấp

4.2.2 Thiết kế lớp

- Chi tiết lớp **MyListHomeScreen**

**Hình 4.10:** Sơ đồ lớp myListHomeScreen

Thuộc tính

#	Tên	Kiểu dữ liệu	Mô tả
1	summaryData	object	Thông tin tóm lược tổng hợp từ số liệu toàn mã cổ phiếu có trong danh mục
2	transactionsGroupedByStockCode	object	Các giao dịch được nhóm theo mã cổ phiếu
3	listData	array	Dữ liệu mục Danh mục
4	rankData	array	Dữ liệu tính toán cho mục Xếp hạng
5	rankGroupScore	object	Điểm số trên toàn danh mục
6	basicData	array	Dữ liệu mục Cơ bản
7	analysisData	object	Dữ liệu mục Phân tích
8	analysisTransactionHistory	array	Dữ liệu phân tích lịch sử
9	selectedMyList	object	Danh mục đang được chọn
10	selectedMyListDetail	object	Danh sách giao dịch của danh mục
11	allMyLists	array	Danh sách tất cả danh mục

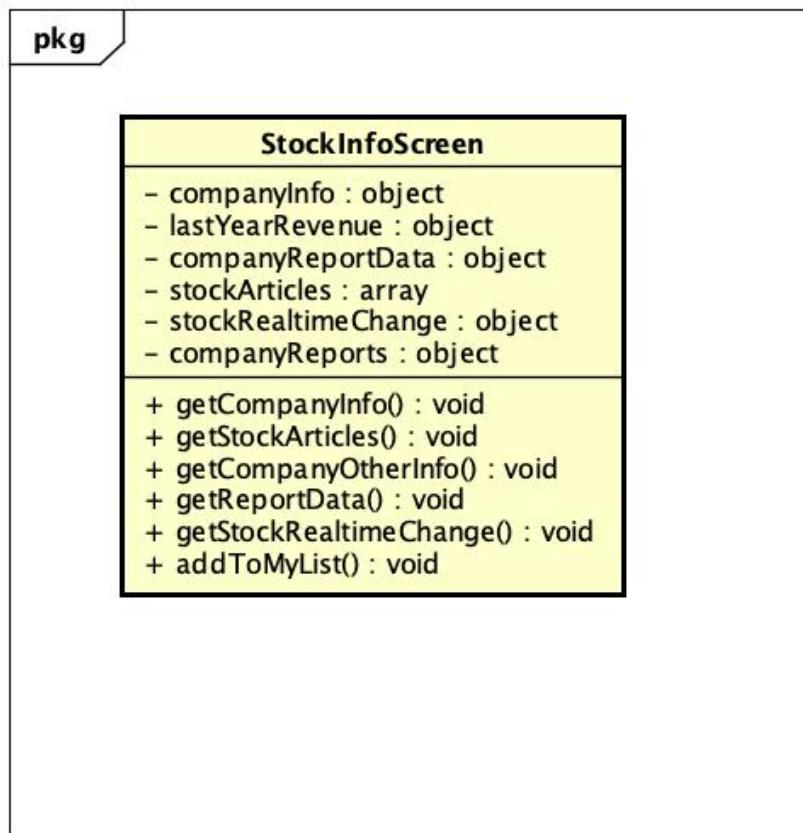
Bảng 4.1: Thuộc tính của myListHomeScreen

Phương thức

#	Tên	Tham số	Kiểu dữ liệu trả về	Mô tả
1	getMyListData	Không có	void	Lấy dữ liệu lịch sử giao dịch, dữ liệu cho từng mục của danh mục đang được chọn
2	getAnalysis TransactionHistoryData	Không có	void	Lấy lịch sử giao dịch của danh mục
3	handlePress ImportTransaction	Không có	void	Nhập dữ liệu từ file excel
4	handlePress AddTransaction	Không có	void	Nhập dữ liệu

Bảng 4.2: Phương thức của myListHomeScreen

- **Chi tiết lớp StockInfoScreen**



Hình 4.11: Sơ đồ lớp StockInfoScreen

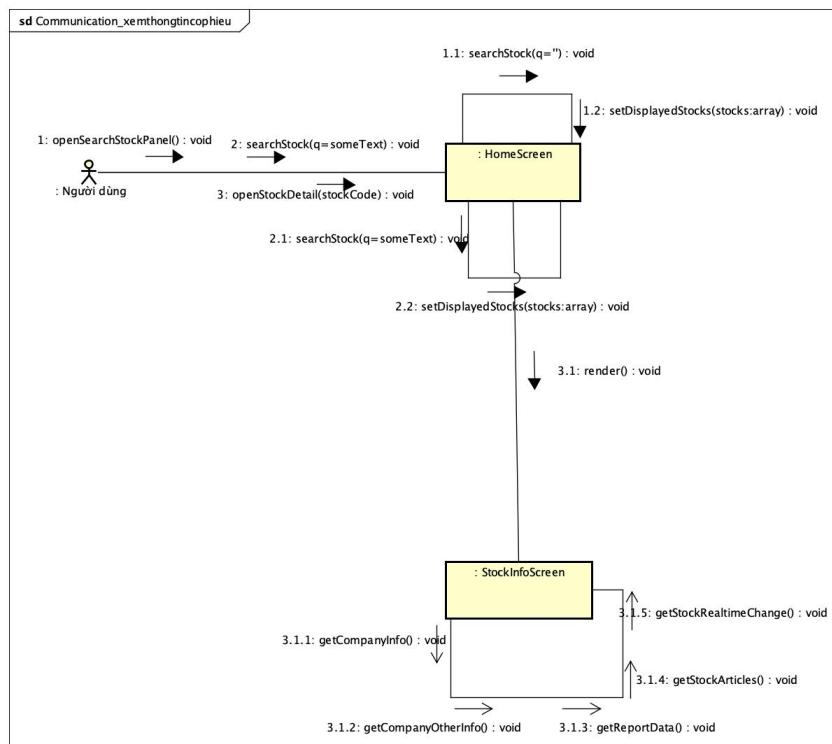
Thuộc tính

#	Tên	Kiểu dữ liệu	Mô tả
1	companyInfo	object	Thông tin cơ bản của công ty phát hành
2	lastYearRevenue	object	Doanh thu năm gần nhất
3	companyReportData	object	Báo cáo tài chính gần nhất
4	stockArticles	array	Danh sách các bài viết liên quan
5	stockRealtimeChange	object	Thông tin cổ phiếu tại hiện tại
6	companyReports	object	Báo cáo tài chính theo mẫu

Bảng 4.3: Thuộc tính của StockInfoScreen***Phương thức***

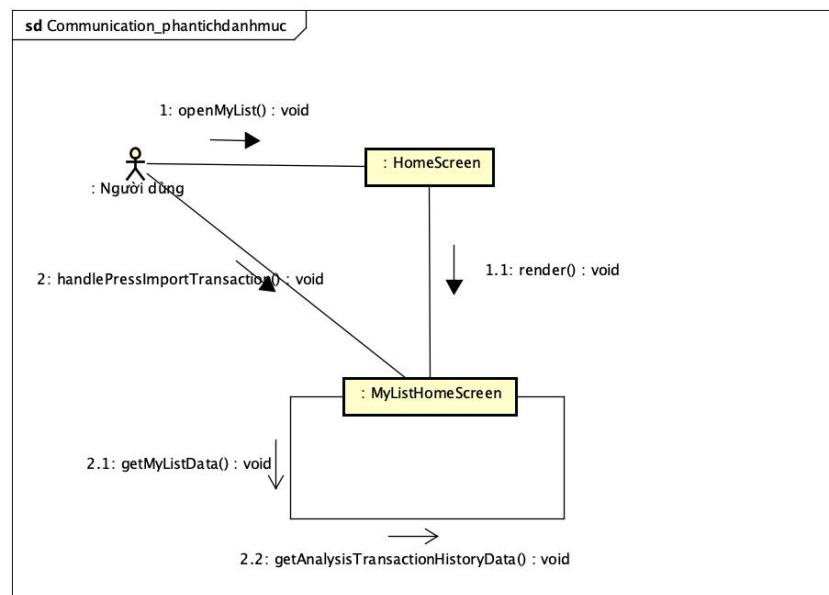
#	Tên	Tham số	Kiểu dữ liệu trả về	Mô tả
1	getCompanyInfo	Không	void	Lấy thông tin công ty
2	getStockArticles	Không	void	Lấy danh sách các bài viết liên quan
3	getCompanyOtherInfo	Không	void	Lấy thông tin liên quan
4	getReportData	Không	void	Lấy báo cáo tài chính theo mẫu
5	getStockRealtimeChange	Không	void	Lấy dữ liệu cổ phiếu thời gian thực
6	addToMyList	Không	void	Chuyển đến màn thêm giao dịch vào danh mục

Bảng 4.4: Phương thức của StockInfoScreen***Biểu đồ giao tiếp của use case Xem thông tin mã chứng khoán***



Hình 4.12: Biểu đồ giao tiếp của use case Xem thông tin mã chứng khoán

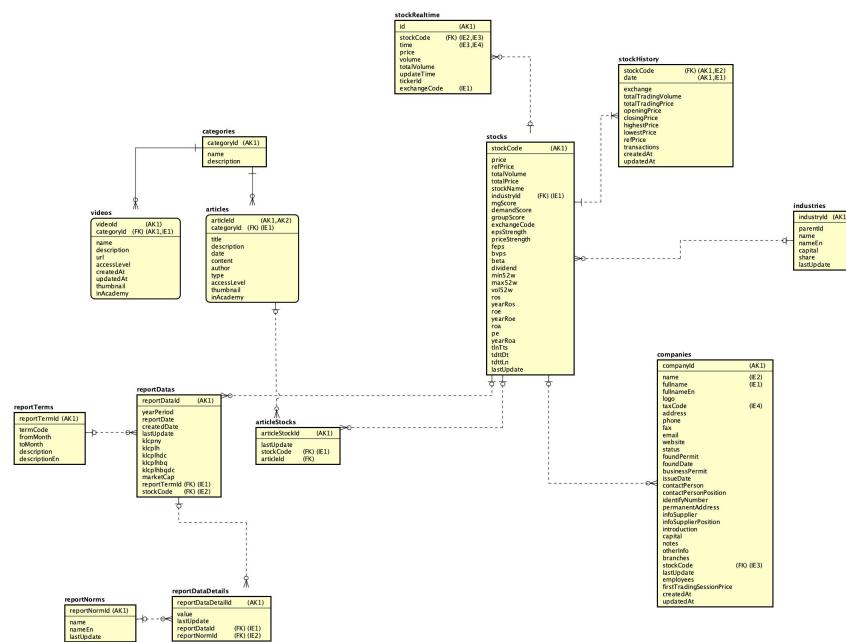
Biểu đồ giao tiếp của use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán



Hình 4.13: Biểu đồ giao tiếp của use case Xem dữ liệu tính toán từ lịch sử giao dịch chứng khoán

4.2.3 Thiết kế cơ sở dữ liệu

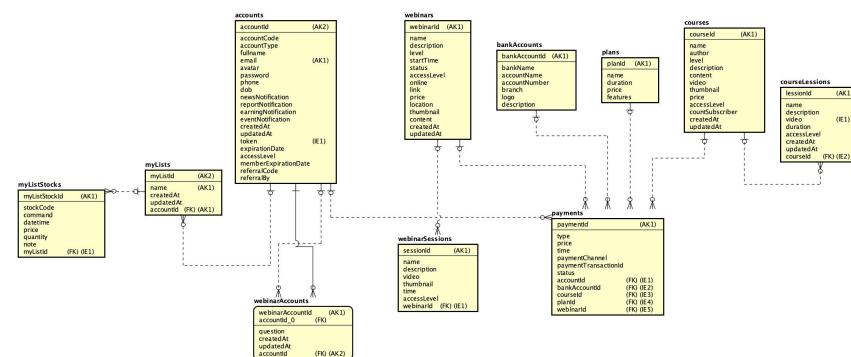
Sơ đồ thực thể liên kết



Hình 4.14: Sơ đồ thực thể liên kết - Liên quan đến cổ phiếu

Mô tả:

- stocks**: Thông tin cơ bản về cổ phiếu như mã chứng khoán, giá hiện tại, mã ngành niêm yết ...
- stockRealtime**: Các giao dịch diễn ra của cổ phiếu, được cập nhật liên tục thời gian thực
- stockHistory**: Lưu thông tin cuối phiên giao dịch của ngày của cổ phiếu
- industries**: Các nhóm ngành của cổ phiếu
- companies**: Công ty phát hành cổ phiếu
- reportDatas**: Báo cáo của công ty phát hành cổ phiếu
- reportTerms**: Các định kỳ về thời gian phát hành báo cáo
- reportDataDetails**: Số liệu có trong một báo cáo về một thông tin nào đó
- reportNorms**: Loại thông tin có trong báo cáo
- articles**: Bài viết
- articleStocks**: Lưu quan hệ giữa bài viết và mã cổ phiếu
- videos**: Video
- categories**: Thể loại của video, bài viết



Hình 4.15: Sơ đồ thực thể liên kết - Liên quan đến tài khoản

Mô tả:

- **accounts**: Tài khoản
- **myLists**: Danh mục của các người dùng
- **myListStocks**: Giao dịch của người dùng trong từng danh mục
- **webinars**: Hội thảo về chứng khoán
- **webinarSession**: Các phần của hội thảo
- **courses**: Khoá học về chứng khoán
- **courseLessons**: Các bài học của khoá học
- **plans**: Gói thành viên
- **payments**: Giao dịch của người dùng với gói thành viên, hội thảo hoặc khoá học
- **bankAccounts**: Tài khoản ngân hàng hưởng thụ khi người dùng thực hiện giao dịch
- **webinarAccounts**: Lưu quan hệ người dùng tham gia hội thảo nào, với hội thảo online

Thiết kế cơ sở dữ liệu với MySQL



Hình 4.16: Sơ đồ cơ sở dữ liệu

Mô tả:

- stocks

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"stockCode"	varchar(16)	Mã chứng khoán
2			"price"	float	Giá giao dịch hiện tại
3			"refPrice"	float	Giá giao dịch trước đó
4			"totalVolume"	bigint	Tổng khối lượng giao dịch gần nhất
5			"totalPrice"	bigint	Tổng giá trị giao dịch gần nhất
6			"stockName"	varchar(256)	Tên đầy đủ
7	X		"industryId"	int	Mã ngành
8			"mgScore"	float	Điểm mg
9			"demandScore"	varchar(32)	A+, A, B+
10			"groupScore"	float	Điểm nhóm
11			"exchangeCode"	varchar(16)	Sàn chứng khoán
12			"epsStrength"	float	Sức mạnh eps
13			"priceStrength"	float	Sức mạnh giá
14			"feps"	float	
15			"byps"	float	
16			"beta"	float	
17			"dividend"	float	
18			"min52w"	float	Giá thấp nhất trong 52 tuần
19			"max52w"	float	Giá cao nhất trong 52 tuần
20			"vol52w"	bigint	Khối lượng giao dịch cao nhất trong 52 tuần
21			"ros"	float	
22			"yearRos"	float	
23			"roe"	float	
24			"yearRoe"	float	
25			"roa"	float	
26			"pe"	float	
27			"yearRoa"	float	
28			"tlnTts"	float	
29			"tdttDt"	float	
30			"tdttLn"	float	

- **stockRealtime**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"id"	int	
2		X	"stockCode"	varchar(16)	Mã cổ phiếu
3			"time"	datetime	Thời gian giao dịch
4			"price"	float	Giá giao dịch
5			"volume"	bigint	Khối lượng giao dịch
6			"totalVolume"	bigint	Tổng khối lượng giao dịch đến thời điểm hiện tại
7			"updateTime"	datetime	Thời gian cập nhật
8			"tickerId"	bigint	
9			"exchangeCode"	varchar(16)	Sàn giao dịch

- **stockHistory**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X	X	"stockCode"	varchar(16)	Mã cổ phiếu
2	X		"date"	date	Ngày ghi sổ
3			"exchange"	varchar(16)	Sàn chứng khoán
4			"totalTradingVolume"	bigint	Tổng khối lượng giao dịch
5			"totalTradingPrice"	bigint	Tổng giá trị giao dịch
6			"openingPrice"	float	Giá mở cửa
7			"closingPrice"	float	Giá đóng cửa
8			"highestPrice"	float	Giá cao nhất
9			"lowestPrice"	float	Giá thấp nhất
10			"refPrice"	float	Chênh lệch giá

- **industries**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"industryId"	int	Mã định danh
2		X	"parentId"	int	Mã ngành mẹ
3			"name"	varchar(256)	Tên
4			"nameEn"	varchar(256)	Tên tiếng Anh
5			"capital"	double	Vốn hoá
6			"share"	float	

- **companies**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"companyId"	int	Định danh
2			"name"	varchar(256)	Tên công ty
3			"fullname"	varchar(512)	Tên đầy đủ
4			"fullnameEn"	varchar(512)	Tên đầy đủ tiếng Anh
5			"logo"	varchar(1024)	Logo công ty
6			"taxCode"	varchar(32)	Mã số thuế
7			"address"	varchar(512)	Địa chỉ
8			"phone"	varchar(256)	SĐT
9			"fax"	varchar(64)	Số fax
10			"email"	varchar(64)	Email
11			"website"	varchar(1024)	Website
12			"status"	tinyint	Tình trạng (Đã niêm yết trên sàn giao dịch hay chưa)
13			"foundPermit"	varchar(64)	Quyết định thành lập
14			"foundDate"	date	Ngày thành lập
15			"businessPermit"	varchar(64)	Giấy phép kinh doanh
16			"issueDate"	date	Ngày phát hành
17			"contactPerson"	varchar(128)	Người liên lạc
18			"contactPersonPosition"	varchar(64)	Chức vụ người liên lạc
19			"identifyNumber"	varchar(64)	Số căn cước
20			"permanentAddress"	varchar(1024)	Trụ sở
21			"infoSupplier"	varchar(128)	Người cung cấp thông tin
22			"infoSupplierPosition"	varchar(128)	Chức vụ của người cung cấp thông tin
23			"introduction"	mediumtext	Giới thiệu
24			"capital"	bigint	Vốn hoá niêm yết
25			"notes"	mediumtext	Ghi chú
26			"otherInfo"	mediumtext	Thông tin phụ
27	X		"industryId"	int	Mã ngành niêm yết
28			"branches"	mediumtext	
29			"stockCode"	varchar(8)	
30			"employees"	int	Số nhân viên
31			"firstTradingSessionPrice"	varchar(256)	Giá khi niêm yết

- reportDatas

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"reportDataId"	int	
2		X	"stockCode"	varchar(16)	Mã cổ phiếu
3		X	"reportTermId"	int	Mã thời gian
4			"yearPeriod"	int	Năm báo cáo
5			"reportDate"	date	Ngày báo cáo
6			"createdDate"	time	Ngày tạo bản ghi
7			"lastUpdate"	time	Ngày cập nhật bản ghi
8			"klcpny"	bigint	Khối lượng cổ phiếu niêm yết
9			"klcplh"	bigint	Khối lượng cổ phiếu lưu hành
10			"klcplhdc"	bigint	
11			"klcplhbq"	bigint	Khối lượng cổ phiếu lưu hành bình quân
12			"klcplhbqdc"	bigint	
13			"marketCap"	bigint	Giá trị trên thị trường

- **reportTerms**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"reportTermId"	int	Định danh
2			"termCode"	varchar(256)	Mã kỳ
3			"fromMonth"	int	Từ tháng
4			"toMonth"	int	Đến tháng
5			"description"	varchar(256)	Mô tả
6			"descriptionEn"	varchar(256)	Mô tả tiếng Anh

- **reportDataDetails**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"reportDataDetailId"	int	Định danh
2		X	"reportDataId"	int	Mã báo cáo
3		X	"reportNormId"	int	Mã loại thông tin
4			"value"	double	Giá trị

- **reportNorms**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"reportNormId"	int	Định danh
2			"name"	varchar(256)	Tên
3			"nameEn"	varchar(256)	Tên tiếng Anh

- **articles**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"articleId"	int	Định danh
2		X	"categoryId"	int	Mã thể loại
3			"title"	varchar(1024)	Tiêu đề
4			"description"	mediumtext	Mô tả
5			"date"	datetime	Ngày ghi trên bài viết
6			"content"	longtext	Nội dung
7			"author"	varchar(128)	Tác giả
8			"type"	varchar(16)	'ARTICLE RESEARCH'
9			"accessLevel"	tinyint	Cấp độ truy cập
10			"thumbnail"	varchar(1024)	Ảnh thumbnail

- **articleStocks**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"articleStockId"	int	Định danh
2		X	"articleId"	int	Mã bài viết
3		X	"stockCode"	varchar(256)	Mã cổ phiếu

- **videos**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"videoId"	int	Định danh
2			"name"	varchar(128)	Tên
3			"description"	varchar(1024)	Mô tả
4			"url"	varchar(1024)	Đường dẫn
5		X	"categoryId"	int	Mã thể loại
6			"accessLevel"	tinyint	Cấp độ truy cập
7			"thumbnail"	varchar(256)	Ảnh thumbnail

- **categories**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1			"categoryId"	int	Định danh
2	X		"name"	varchar(256)	Tên thể loại
3			"description"	varchar(1024)	Mô tả

- **accounts**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"accountId"	int	Định danh
2			"email"	varchar(128)	Email
3			"avatar"	varchar(1024)	Ảnh đại diện
4			"password"	varchar(256)	Mật khẩu
5			"phone"	varchar(64)	SĐT
6			"dob"	datetime	Ngày sinh
7			"token"	varchar(256) Token	
8			"expirationDate"	datetime	Thời hạn token
9			"accessLevel"	int	Cấp độ truy cập 0: Quản trị viên 1: Người dùng còn hạn thành viên 2: Người dùng thường
10			"member ExpirationDate"	datetime	Thời hạn gói thành viên

- **myLists**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"myListId"	int	Định danh
2			"name"	varchar(256)	Tên danh mục
3		X	"accountId"	int	Mã chủ danh mục

- **myListStocks**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"myListStockId"	int	Định danh
2		X	"myListId"	int	Mã danh mục
3		X	"stockCode"	varchar(16)	Mã cổ phiếu
4			"command"	enum	Lệnh giao dịch BUY SELL
5			"datetime"	datetime	Ngày giao dịch
6			"price"	int	Giá giao dịch (1 đơn vị cổ phiếu)
7			"quantity"	int	Khối lượng giao dịch

- **webinars**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"webinarId"	int	Định danh
2			"name"	varchar(256)	Tên hội thảo
3			"description"	text	Mô tả
4			"level"	varchar(32)	Cấp độ hội thảo
5			"startTime"	datetime	Thời gian bắt đầu
6			"status"	varchar(16)	Tình trạng hội thảo
7			"accessLevel"	tinyint	Cấp độ truy cập
8			"online"	tinyint	Có phải hội thảo online hay không
9			"link"	varchar(1024)	Đường dẫn (nếu là hội thảo online)
10			"price"	int	Giá hội thảo
11			"location"	varchar(1024)	Địa điểm
12			"thumbnail"	varchar(1024)	Ảnh thumbnail
13			"content"	text	Nội dung tóm tắt

- **webinarSession**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"sessionId"	int	Định danh
2		X	"webinarId"	int	Mã hội thảo
3			"name"	varchar(256)	Tên hội thảo
4			"description"	text	Mô tả hội thảo
5			"video"	varchar(1024)	Video giới thiệu
6			"thumbnail"	varchar(1024)	Ảnh thumbnail
7			"time"	datetime	Thời gian
8			"accessLevel"	tinyint	Cấp độ truy cập

- **courses**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1			"courseId"	int	Định danh
2			"name"	varchar(256)	Tên
3			"author"	varchar(256)	Diễn giả
4			"level"	int	Cấp độ
5			"description"	text	Mô tả khoá học
6			"content"	text	Nội dung tóm tắt
7			"video"	varchar(1024)	Video giới thiệu
8			"thumbnail"	varchar(1024)	Ảnh thumbnail
9			"price"	float	Giá khoá học
10			"accessLevel"	int	Cấp độ truy cập
11			"countSubscriber"	int	Số lượng người đăng ký

- courseLessons

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"lessonId"	int	Định danh
2		X	"courseId"	int	Mã khoá học
3			"name"	varchar(256)	Tên bài học
4			"description"	varchar(1024)	Mô tả bài học
5			"video"	varchar(512)	Video bài học
6			"duration"	int	Thời lượng (giây)
7			"accessLevel"	tinyint	Cấp độ truy cập

- plans

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"planId"	int	Định danh
2			"name"	varchar(128)	Tên gói
3			"duration"	smallint	Thời hạn (tháng)
4			"price"	float	Giá bán
5			"features"	json	Tính năng

- payments

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"paymentId"	int	Định danh
2		X	"accountId"	int	Mã tài khoản thanh toán
3			"type"	varchar(32)	Loại thanh toán COURSE MEMBER WEBINAR
4		X	"courseId"	int	Mã khoá học
5		X	"planId"	int	Mã gói thành viên
6			"webinarId"	int	
7			"price"	int	
8			"time"	datetime	
9			"paymentChannel"	varchar(32)	
10			"payment TransactionId"	varchar(1024)	
11			"status"	enum	Tình trạng thanh toán CONFIRMED FAILED PENDING
12		X	"bankAccountId"	int	Mã tài khoản hưởng thụ

- bankAccounts

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"bankAccountId"	int	Định danh
2			"bankName"	varchar(256)	Tên ngân hàng
3			"accountName"	varchar(256)	Tên chủ tài khoản
4			"accountNumber"	varchar(128)	Số tài khoản
5			"branch"	varchar(256)	Chi nhánh
6			"logo"	varchar(1024)	Logo ngân hàng
7			"description"	varchar(512)	Mô tả

- **webinarAccounts**

#	PK	FK	Tên trường	Kiểu dữ liệu	Mô tả
1	X		"webinarAccountId"	int	Định danh
2		X	"webinarId"	int	Mã khóa học
3		X	"accountId"	int	Mã tài khoản
4			"question"	text	Câu hỏi khi đăng ký tham gia

4.3 Xây dựng ứng dụng

4.3.1 Thư viện và công cụ sử dụng

Mục đích	Công cụ	Phiên bản	Địa chỉ URL
IDE lập trình	Visual Studio Code	1.69.2 (Universal)	https://code.visualstudio.com/
Kiểm thử API	Postman	9.22.2	https://www.postman.com/
Ngôn ngữ lập trình	JavaScript		https://developer.mozilla.org/en-US/docs/Web/JavaScript
Trình runtime	Nodejs	16.15.1	https://nodejs.org/en/
Quản lý gói	Yarn	1.22.19	https://yarnpkg.com/
Quản lý CSDL	PhpMyAdmin		https://www.phpmyadmin.net/
Hệ QTCSQL	MySQL		https://www.mysql.com/
Thiết kế giao diện	Figma	116.2.3	https://figma.com/
Phân tích thiết kế	Astah Professional	8.5.0/39c620	https://astah.net/products/astah-professional/

Bảng 4.5: Danh sách thư viện và công cụ sử dụng

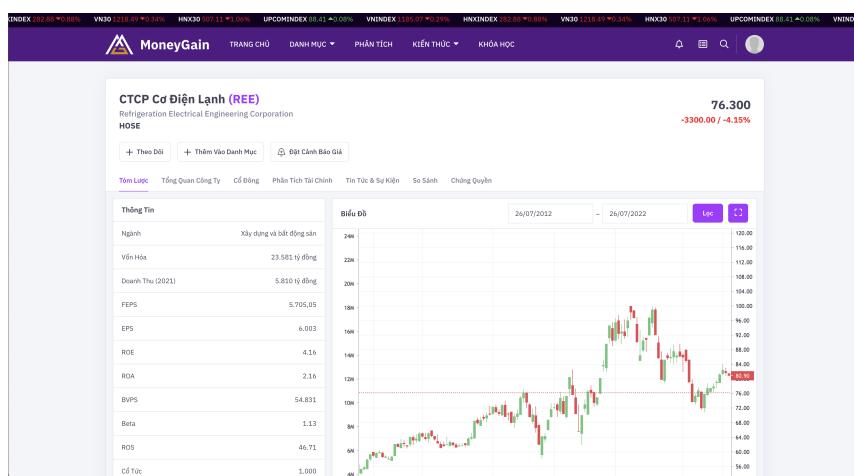
4.3.2 Kết quả đạt được

#	Thông tin	Mô tả
1	Website dành cho người dùng	<ul style="list-style-type: none"> Chức năng: Dành cho người dùng cuối có nhu cầu tìm hiểu và tham gia vào thị trường chứng khoán, muốn có thêm hiểu biết <ul style="list-style-type: none"> Dung lượng khi đóng gói: 3.34 MB Dung lượng mã nguồn: 38.6 MB Môi trường: Trình duyệt web
2	Website cho quản trị viên	<ul style="list-style-type: none"> Chức năng: Dành cho quản trị viên để quản lý nội dung website cho người dùng cuối <ul style="list-style-type: none"> Dung lượng khi đóng gói: 4.3 MB Dung lượng mã nguồn: 21.1 MB Môi trường: Trình duyệt web
3	Server	<ul style="list-style-type: none"> Chức năng: Cung cấp API, cập nhật, tính toán dữ liệu <ul style="list-style-type: none"> Dung lượng mã nguồn: 87.6 MB Môi trường: Máy chủ

Bảng 4.6: Kết quả đạt được

4.3.3 Minh họa các chức năng chính

- Màn hình thông tin cổ phiếu/công ty

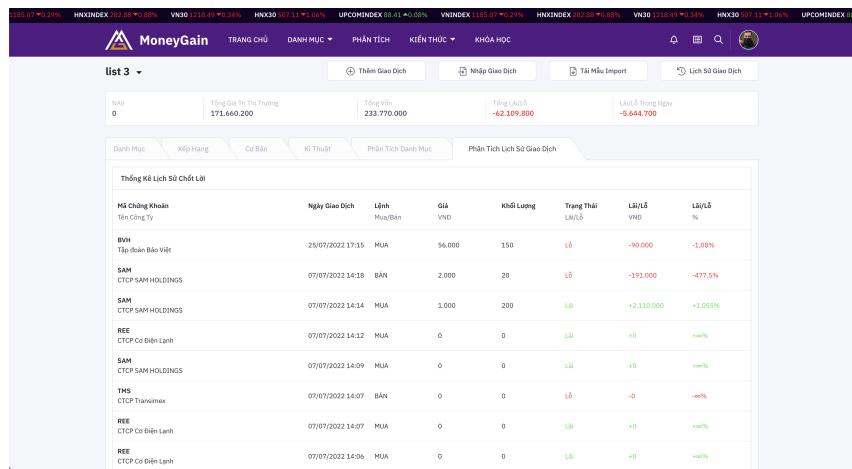


Hình 4.17: Màn hình thông tin công ty

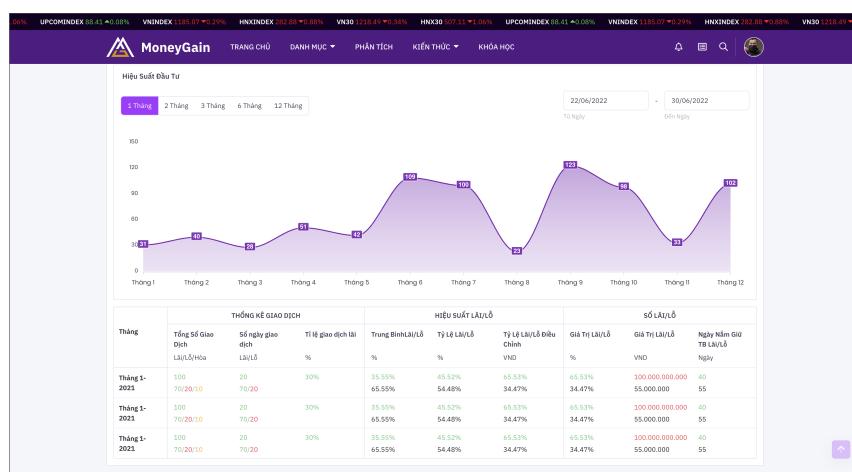
Mô tả: Đưa ra thông tin tổng quan của công ty và thông tin cổ phiếu niêm yết trên sàn giao dịch. Biểu đồ nền thể hiện biến động của mã cổ phiếu với mỗi giao dịch, có thể phóng to ra xem và xem theo từng quãng thời gian

- Màn hình phân tích danh mục

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



Hình 4.18: Phân tích danh mục



Hình 4.19: Phân tích lịch sử giao dịch

Mô tả: Tổng hợp tính toán dựa trên dữ liệu giao dịch của người dùng và thông tin dữ liệu cập nhật liên tục.

- **Màn hình bài viết phân tích**



Hình 4.20: Chi tiết một bài viết



Hình 4.21: Chi tiết một khoá học

Mô tả: Đây là nơi đưa ra bài viết, bài báo liên quan đến cổ phiếu.

- **Khoá học** Mô tả: Khoá học bao gồm nhiều video bài học. Một khoá học có thể có các cấp độ khác nhau từ Cơ bản đến Nâng cao, có thể phải trả phí để truy cập khoá học.

4.4 Triển khai

Máy chủ: Hệ điều hành Xubuntu, bộ nhớ 32GB, dung lượng lưu trữ 2TB

- **Triển khai server** Triển khai thử nghiệm trên máy tính cá nhân và máy chủ cho tốc độ phản hồi API nhanh.
- **Triển khai website** Triển khai thử nghiệm trên máy tính cá nhân và máy chủ cho tốc độ phản hồi API nhanh.
- **Triển khai cơ sở dữ liệu** Cơ sở dữ liệu được lưu trữ tại máy chủ của công ty thực tập, cho tốc độ truy cập nhanh. Việc truy vấn giữ server và cơ sở dữ liệu được thực hiện bên trong cùng một hệ thống.

CHƯƠNG 5. CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT

Trong đồ án này em đã tìm hiểu về các phương thức tính toán về các mã cổ phiếu để đưa ra số liệu cho người chơi chứng khoán đọc.

Việc lấy được dữ liệu thời gian thực, các giao dịch đang được diễn ra trên các sàn giao dịch của Việt Nam được lấy thông qua một socket kết nối đến với bên Công ty cổ phần chứng khoán SSI. Thông tin các công ty được nhập vào CSDL từ các file excel lấy từ trang Vietstock Finance.

Để phục vụ công việc tìm hiểu và làm đồ án em đã tìm hiểu cách đọc các bảng chỉ số như biểu đồ nến của các giao dịch một mã cổ phiếu, các công thức tính toán lãi lỗ, sự tăng giảm điểm và cách so sánh với điểm của thời gian thực.

Một trong những công thức tính toán cần phải xử lý khi được client yêu cầu là tính toán chỉ số P/B của một mã cổ phiếu. P/B được dùng để so sánh giá trị thực và giá trị ghi sổ của một cổ phiếu, từ đó biết được giá cổ phiếu đang cao/thấp bao nhiêu lần so với giá trị ghi sổ. Công thức tính là:

$$P/B = Price / (Book value per share)$$

trong đó Price là giá trị thị trường hiện tại của cổ phiếu, còn Book value per share là giá trị ghi sổ của một cổ phiếu, được tính bằng công thức:

$$\text{Giá trị ghi sổ} = \frac{\text{Tổng giá trị tài sản} - \text{Giá trị tài sản vô hình} - \text{Nợ}}{\text{Số lượng cổ phiếu lưu hành}}$$

Tài sản vô hình của doanh nghiệp là tài sản không có hình thái vật chất và có khả năng tạo ra các quyền, lợi ích kinh tế và phải thỏa mãn đồng thời các điều kiện sau:

- Không có hình thái vật chất; tuy nhiên một số tài sản vô hình có thể chứa đựng trong hoặc trên thực thể vật chất, nhưng giá trị của thực thể vật chất là không đáng kể so với giá trị tài sản vô hình
- Có thể nhận biết được và có bằng chứng hữu hình về sự tồn tại của tài sản vô hình (ví dụ: hợp đồng, bằng chứng nhận, hồ sơ đăng ký, đĩa mềm máy tính, danh sách khách hàng, báo cáo tài chính, v.v.)
- Có khả năng tạo thu nhập cho người có quyền sở hữu

- Giá trị của tài sản vô hình có thể định lượng được

Các thông tin Tổng giá trị tài sản, Giá trị tài sản vô hình, Nợ và Số lượng cổ phiếu lưu hành được truy vấn từ bảng lưu báo cáo tài chính của công ty. Còn Giá trị thị trường hiện tại sẽ được lấy trực tiếp từ bảng lưu thông tin cổ phiếu, thông tin giá trị được cập nhật liên tục dựa trên thông tin có được từ phía SSI.

Dữ liệu cho một mục trong phần Phân tích danh mục (phân tích lịch sử giao dịch của người dùng) đòi hỏi sự kết nối giữa các bảng trong truy vấn sao cho hợp lý và đáp ứng về yêu cầu phi chức năng là thời gian phản hồi, ví dụ như mục Cơ bản dưới đây

Mã Chứng Khoán Tên Công Ty	Giá Thị Trường VND	Tăng Trưởng Doanh Thu Quý Gần Nhất	Tăng Trưởng Lợi Nhuận Quý Gần Nhất	PE	PB	ROE	ROA	Beta
REE CTCP Cơ Điện Lạnh	75.200	-7.21%	-3.31%		2.36	4.16	2.16	1.13
ACB Ngân hàng TMCP Á Châu	24.600	-6.61%	-26.13%		1.45	7.06	0.62	0.88
AAA CTCP Nhựa An Phát Xanh	12.250	+4.25%	-27.58%		0.92	1.63	0.86	1.56
VIC Tập đoàn Vingroup - CTCP	64.000	+89.03%	-1906.31%		2.89	1.67	0.56	0.47
BCE CTCP Xây dựng và Giao thông Bình Dương	9.200	-96.42%	-99.37%		0.08	0.14	0.07	1.42
AG1 CTCP 28.1	6.200	---%	---%			8.42	2.23	0.01
ACV Tổng Công ty Cảng Hàng Không Việt Nam - CTCP	87.000	-54.47%	-61.91%		4.94	2.3	1.61	0.73
BTP CTCP Nhiệt điện Bà Rịa	16.600	+495143.34%	---%		0.80	-0.95	-0.68	0.64
BVH Tập đoàn Bản Việt	57.800	---%	---%		2.32	6.34	0.27	1.22

Hình 5.1: Tab cơ bản trong Danh mục của tôi

Để có thể tính được sự tăng trưởng về doanh thu hay lợi nhuận cần truy vấn dữ liệu báo cáo tài chính của hai quý gần nhất, trong trường hợp doanh nghiệp đó không có dữ liệu phải dùng bản ghi tương tự khác, vì có hai loại dữ liệu doanh thu/lợi nhuận mà doanh nghiệp ghi lại.

Việc quản lý cấp độ truy cập bài viết, khoá học hay hội thảo với người dùng được giải quyết bằng cách gắn một cấp độ truy cập cho đối tượng, và người dùng thuộc cấp độ truy cập đó hoặc cao hơn mới có thể truy cập.

Xử lý gửi mail cho người dùng khi thanh toán, thanh toán được xác nhận em sử dụng thư viện **nodemailer** với OAuth2 để tránh việc sử dụng thẳng password của email. Em đã tạo một ứng dụng trên Google Cloud Console, tại id và refresh token để sử dụng địa chỉ mail.

Kiến trúc của sản phẩm là kiến trúc Flux, như được đề cập trong phần 4.1.1, được phát triển riêng cho các dự án dùng React, dựa trên kiến trúc quen thuộc là

MVC. Em đã tìm hiểu và áp dụng kiến trúc vào trong dự án, cho cả phía website quản trị.

Với MVC chúng ta đã quen với thể hiện của ba nhóm đối tượng là Model, View và Controller, thì chuyển sang Flux ta sẽ có Action, Dispatcher và Store tương ứng với vai trò của Controller. Còn View và Model của MVC tương tự như View trong Flux. Việc hiểu vai trò các thành phần cũng là cốt lõi để mã nguồn được gọn gàng, rõ ràng, dễ chỉnh sửa, thay đổi.

CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1 Kết luận

Sản phẩm website cho người dùng cuối đã hoàn thiện đủ chức năng cần thiết của thiết kế ban đầu và đầy đủ chức năng để trợ giúp người dùng khi tham gia thị trường chứng khoán. Những chức năng như thống kê tính toán lịch sử giao dịch, cung cấp bài viết, cung cấp thông tin chỉ số được tham khảo từ các chức năng tương ứng trên finance.vietstock.vn.

Một hạn chế của ứng dụng đó là việc thanh toán của người dùng chưa được tự động, quản trị viên phải tự xác nhận những giao dịch và việc này sẽ gây mất thời gian nếu lượng giao dịch trở nên quá nhiều. Đồng thời hiện tại chưa có tính năng thông báo giao dịch mới cho quản trị viên.

Vì đây là sản phẩm theo yêu cầu của khách hàng nên được đầu tư nhiều về giao diện người dùng, mặc dù các tính năng rất cơ bản, đây cũng là một điểm trừ lớn với đồ án này.

6.2 Hướng phát triển

- Hoàn thiện:

Sản phẩm đã tương đối hoàn thiện và ổn định về mặt tính năng theo yêu cầu và nhu cầu của người dùng. Về phần biểu đồ theo dõi trong thời gian tới sẽ hoàn thiện thêm biểu đồ chỉ số ngành

- Hướng phát triển:

Hiện tại server có thể gửi thông báo cho người dùng qua hai kênh thông tin là hộp thư điện tử và webpush phía server. Nhưng theo thói quen của đa số mọi người, bao gồm cả chính em, là từ chối quyền thông báo của ứng dụng web. Điều đó sẽ làm thông tin cập nhật đến người dùng chỉ còn qua email, gây bất lợi về thời gian và cũng dễ bị trôi thông báo, và cũng không thể gửi nhiều được vì sẽ dễ vào hộp thư spam.

Giải pháp khả dĩ và đơn giản nhất đó là tạo một socket với server để biết khi nào có thông báo mới và hiển thị bằng giao diện của chính website. Khi không mở ứng dụng thì vẫn sẽ nhận thông báo, hoặc không nếu bị từ chối, qua web-push.

Việc cập nhật lịch sử giao dịch là do người dùng cung cấp nên khá bị động. Trong tương lai sẽ cần phải nghiên cứu phương thức xác thực với các ứng dụng

giao dịch chứng khoán phổ biến và sẽ cập nhật giao dịch tự động cho người dùng.

6.3 Lời kết

Trên đây là toàn bộ nội dung đồ án tốt nghiệp của em. Trong quá trình thực hiện còn nhiều sai sót mong quý thầy cô châm trước và chỉ ra cho em để em có thể hoàn thiện sản phẩm cũng như hoàn thiện bản thân mình hơn. Xin cảm ơn thầy cô và mọi người đã dành thời gian cho quyển đồ án này.