

Trường Đại học Bách Khoa Hà Nội
Viện Công nghệ Thông Tin và Truyền Thông

Đồ án Tốt nghiệp Đại học

Xây dựng ứng dụng tư vấn và
quản lý cửa hàng thời trang

Đỗ Đức Thanh

Hà Nội, 06/2018

Trường Đại học Bách Khoa Hà Nội
Viện Công nghệ Thông Tin và Truyền Thông

Đồ án Tốt nghiệp Đại học

Xây dựng ứng dụng tư vấn và
quản lý cửa hàng thời trang

Sinh viên thực hiện **Đỗ Đức Thanh**

Người hướng dẫn **PGS-TS. Cao Tuấn Dũng**

[Sau khi in quyển, sinh viên xin chữ ký của
giáo viên hướng dẫn ở đây. Chữ ký là xác
nhận đồng ý cho phép bảo vệ của giáo viên
hướng dẫn]

Hà Nội, 06/2018

Họ và tên sinh viên: Đỗ Đức Thanh

Điện thoại liên lạc: 01655871797 . Email: doducthanh95@gmail.com

Lớp: CNTT2.03 K58. Hệ đào tạo: đại học

Tôi Đỗ Đức Thanh cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của PGS-TS Cao Tuấn Dũng. 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. Mọi tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ 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 chân thành cảm ơn các thầy cô trong viện đã truyền đạt cho em những kiến thức và kinh nghiệm quý giá trong suốt 5 năm học qua để em có đủ kiến thức để hoàn thành đồ án này.

Em xin cảm ơn bộ môn Công nghệ Phần mềm, viện Công nghệ thông tin và Truyền thông đã tạo điều kiện để em được thực hiện đồ án này.

Em xin gửi lời cảm ơn sâu sắc đến thầy Cao Tuấn Dũng đã hướng dẫn, chỉ dạy chúng em tận tình trong suốt quá trình thực hiện đồ án này.

Cuối cùng, em xin được gửi lời cảm ơn chân thành đến gia đình bạn bè đã luôn tận tâm giúp đỡ, động viên, đóng góp ý kiến giúp em hoàn thành đồ án tốt nghiệp thành công.

Tóm tắt

Ngày nay với sự phát triển nhanh chóng về kinh tế, các lĩnh vực kinh doanh cũng có sự phát triển vượt bậc. Thời trang vốn từ lâu đã là một trong những thành phần chủ đạo trong các nền kinh tế cũng không nằm trong vùng ngoại lệ. Ngoài những công ty, những cửa hàng lớn đã có từ trước, các đơn vị kinh doanh nhỏ lẻ, kinh doanh online cũng ngày một phát triển, nâng tầm về quy mô và số lượng. Thị trường kinh doanh thời trang vốn đã sôi động nay càng trở thành mảnh đất giàu tiềm năng cho các nhà đầu tư hơn.

Khi mở rộng quy mô kinh doanh, không chỉ các công ty, cửa hàng lớn mà ngay cả những hộ tiểu thương, kinh doanh online cũng đã và đang gặp những khó khăn trong việc quản lý, nắm bắt thông tin về tất cả sản phẩm trong cửa hàng của mình. Các chủ cửa hiệu cũng mất rất nhiều thời gian quản lý, cập nhật tình hình kinh doanh của các cửa hàng. Mặc dù một số nhãn hàng thời trang đã tự mình sản xuất các ứng dụng tra cứu và quản lý cho mình, song không phải cửa hàng nào cũng có điều kiện như vậy. Vì thế nhu cầu về quản lý và tư vấn vẫn còn rất lớn, và còn một lượng lớn khách hàng tiềm năng.

Nắm bắt được các vướng mắc đó, em đã đề xuất thực hiện ứng dụng tư vấn và quản lý bán hàng nhằm giúp các nhân viên tư vấn sản phẩm một cách hiệu quả, các chủ cửa hiệu cập nhật tình hình kinh doanh nhanh chóng. Các nhân viên bán hàng có thể tra cứu thông tin sản phẩm qua màu sắc, tên gọi, mã vạch, các sản phẩm bán chạy, khuyến mại,... Chủ cửa hiệu có thể nắm bắt được tình hình kinh doanh, doanh thu của các cửa hàng, phân tích xu hướng từ những tìm kiếm lựa chọn của người dùng để đưa ra chiến lược kinh doanh và nhập hàng hợp lý.

Bố cục đồ án chia thành 6 chương:

- Chương 1: Giới thiệu đề tài
- Chương 2: Khảo sát và phân tích yêu cầu
- Chương 3: Công nghệ sử dụng
- Chương 4: Phát triển và triển khai ứng dụng
- Chương 5: Các giải pháp và đóng góp nổi bật
- Chương 6: Kết luận và hướng phát triển

Mục lục

Danh mục hình vẽ.....	x
Danh mục bảng.....	xi
Danh mục các từ viết tắt.....	xiii
Danh mục thuật ngữ.....	xiv
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	2
1.4 Bố cục đồ án	2
Chương 2 Khảo sát và phân tích yêu cầu	4
2.1 Khảo sát hiện trạng.....	4
2.2 Tổng quan chức năng.....	5
2.2.1 Biểu đồ use case tổng quan	5
2.3 Đặc tả Usecase chức năng.....	7
2.3.1 Đặc tả use case đăng nhập	7
2.3.2 Đặc tả use case tìm kiếm sản phẩm theo tên	8
Hình 3: Đặc tả usecase tìm kiếm sản phẩm theo tên	9
2.3.6 Đặc tả use case tra cứu hàng mới về	9

2.3.5	Đặc tả use case tra cứu hàng bán chạy	10
	Hình 5: Đặc tả use case tra cứu hàng bán chạy	11
2.3.6	Đặc tả use case tra cứu hàng khuyến mại	11
2.3.7	Đặc tả use case tra cứu hàng hoá tìm kiếm nhiều	12
2.3.8	Đặc tả use case quản lý doanh thu cửa hàng	13
2.3.9	Đặc tả use case quản lý sản phẩm	13
2.4	Yêu cầu phi chức năng.....	14
Chương 3 Công nghệ sử dụng 15		
3.1	Microservice	15
3.2	Nodejs.....	17
3.3	Swift	18
3.4	MySQL	19
Chương 4 Phát triển và triển khai ứng dụng 22		
4.1	Thiết kế kiến trúc	22
4.1.1	Lựa chọn kiến trúc phần mềm	22
4.1.2	Thiết kế tổng quan	26
4.2	Thiết kế chi tiết	27
4.2.1	Thiết kế giao diện	27
4.2.2	Thiết kế lớp.....	30
4.2.3	Thiết kế cơ sở dữ liệu	34
4.3	Xây dựng ứng dụng.....	36
4.3.1	Thư viện và công cụ sử dụng.....	36
4.3.2	Kết quả đạt được.....	37
4.3.3	Minh hoạ các chức năng chính	38
4.4	Kiểm thử	38
4.5	Triển khai.....	38
Chương 5 Các giải pháp và đóng góp nổi bật 39		
5.1	Những đóng góp của đề tài trong thực tế.....	39

5.2 Những khó khăn và giải pháp đặt ra trong ĐATN	42
5.2.1 Giải quyết vấn đề tối ưu truy vấn.	42
5.2.2 giải quyết vấn đề bảo mật	43
Chương 6 Kết luận và hướng phát triển 46	
6.1 Kết luận	46
6.2 Hướng phát triển	47
Tài liệu tham khảo 48	

Danh mục hình vẽ

Hình 1 Ví dụ biểu đồ phụ thuộc gói	26
Hình 2 Ví dụ thiết kế gói	Error! Bookmark not defined.
Hình 3 Ví dụ hình vẽ	A-4

Danh mục bảng

Bảng 1 Danh sách thư viện và công cụ sử dụng..... 36

Bảng 2 Ví dụ sử dụng bảng A-3

Danh mục các từ viết tắt

API	Application Programming Interface Giao diện lập trình ứng dụng
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
CNTT	Công nghệ thông tin
ĐATN	Đồ án tốt nghiệp
SV	Sinh viên
JWT	Json web token

Danh mục thuật ngữ

Browser	Trình duyệt
Cache memory	Bộ nhớ đệm
E-commerce	Thương mại điện tử
Bloatware	Ứng dụng nhà sản xuất tích hợp vào thiết bị
Interpreter	Trình thông dịch
Compiler	Trình biên dịch
Client	Khách hàng, người sử dụng
Server	

Chương 1 Giới thiệu đề tài

1.1 Đặt vấn đề

Từ xưa đến nay, thời trang hay buôn bán quần áo vẫn luôn chiếm một tỉ trọng lớn trong ngành công thương của mỗi quốc gia, mỗi thời kì. Quần áo luôn là một mặt hàng thiết yếu không thể thiếu.

Ngày nay khi đời sống xã hội con người được cải thiện, nhu cầu ăn uống ăn mặc cũng tăng cao. Không chỉ có thế, các yêu cầu về sản phẩm của người tiêu dùng cũng khắt khe hơn trước. Đáp ứng lại sự thay đổi đó, các công ty chuyên về sản xuất thời trang cũng luôn luôn đưa ra thị trường rất nhiều sản phẩm đa dạng cả về mẫu mã và giá cả. Thị trường ngày nay không còn là sân chơi của những công ty kinh doanh lớn hay những cửa hàng có tiếng trước đó nữa mà còn có sự góp mặt của rất nhiều hộ kinh doanh nhỏ lẻ, kinh doanh online. Các cửa hàng chi nhánh thời trang cũng đua nhau mọc lên phủ kín những con phố.

Số lượng cửa hàng, sản phẩm tăng lên kéo theo bài toán về nắm bắt và quản lí thông tin. Việc nắm bắt tình trạng của tất cả các mặt hàng gần như là không thể và là vấn đề lớn đối với các nhân viên của cửa hàng

1.2 Mục tiêu và phạm vi đề tài

Với sự phát triển mạnh mẽ của công nghệ thông tin và truyền thông, sự bùng nổ của các thiết bị di động, cùng với bài toán đặt ra như trên, chúng ta nghĩ ngay đến xây dựng một ứng dụng chạy trên nền tảng mobile.

Ứng dụng này phải đảm bảo hai tiêu chí chính. Thứ nhất là giải quyết được hết vấn đề của bài toán trên. Thứ hai là phải giúp giảm chi phí cho dịch vụ, mang lại lợi ích cho cả công ty và khách hàng của họ. Mặt khác, chi phí cho xây dựng ứng dụng, cũng như triển khai, bảo trì phải thấp.

1.3 Định hướng giải pháp

Trong bài toán này, em định hướng thiết kế một sản phẩm dành cho mobile. Lý do em chọn phát triển ứng dụng cho thiết bị mobile vì hiện nay và tương lai sau này, con người chúng ta sẽ giành chủ yếu thời gian làm việc thông qua các thiết bị di động. Sự tiết kiệm về chi phí, cài đặt nhanh chóng, dễ sử dụng của các thiết bị di động cũng là một trong những ưu điểm khiến em chọn mobile là nền tảng phát triển cho mình.

Ứng dụng sẽ bao gồm 2 phần: server và client. Server có nhiệm vụ lưu trữ các dữ liệu của từng cửa hàng đăng ký. Phía client sẽ gọi và lấy dữ liệu hiển thị thông qua các API. Client cũng gửi các thói quen tìm kiếm của khách hàng lên server để có thể phục vụ cho các quản lý cửa hàng phân tích và đưa ra chiến lược kinh doanh phù hợp.

1.4 Bố cục đồ án

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

Đồ án tốt nghiệp này được tổ chức như sau:

Chương 1 trình bày giới thiệu đề tài cũng như các vấn đề tổng quan. Trong chương này, em trình bày nội dung chính về tên đề tài, lí do chọn đề tài cũng như một số giải pháp định hướng cho đề tài được chọn.

Chương 2 trình bày về khảo sát và phân tích các yêu cầu của khách hàng, tìm hiểu về quy trình nghiệp vụ bán hàng. Trong chương này, nội dung khảo sát và phân tích yêu cầu khách hàng được làm rõ. Các biểu đồ tổng quan về mặt chức năng cũng được đề cập đến trong chương này.

Chương 3 trình bày về các công nghệ và lý thuyết để sử dụng giải quyết bài toán đặt ra. Trong chương này em sẽ trình bày chủ yếu lý thuyết tổng quan của các công nghệ được, thư viện hỗ trợ được sử dụng trong ĐATN.

Chương 4 trình bày về quy trình phát triển và triển khai ứng dụng.

Chương 5 em xin trình bày về giải pháp và các đóng góp cụ thể của bài toán. Chương này sẽ tập trung làm rõ tác dụng của đề tài với người sử dụng thực tế, các khó khăn và hướng giải quyết trong đề tài.

Chương 2 Khảo sát và phân tích yêu cầu

2.1 Khảo sát hiện trạng

Khảo sát chi tiết về yêu cầu của phần mềm sẽ phát triển từ một số trong những nguồn chính sau đây:

- Người dùng, khách hàng
- Các hệ thống đã có
- Các ứng dụng tương tự

Phân tích các yêu cầu đó, từ đó rút ra kết luận cho các tính năng của phần mềm sẽ phát triển.

Qua quá trình tìm hiểu các nhu cầu của các cửa hàng thời trang hiện nay về một sản phẩm có thể hỗ trợ trong việc tư vấn và quản lý bán hàng, em đã thu được các nội dung sau:

- Đối với các chủ cửa hàng thời trang, họ rất quan tâm đến vấn đề tiếp thị, tư vấn khách hàng. Đối với một lượt bán hàng thành công, thời gian họ giành để chăm sóc, tư vấn cho khách hàng của mình là rất lớn. Vào mỗi đợt cao điểm trong ngày, lượng khách đông khiến họ càng khó khăn hơn trong công việc. Việc một cửa hàng có nhiều chi nhánh khiến cho chủ cửa hàng gặp vấn đề trong việc kiểm soát doanh thu và tình hình kinh doanh. Họ không thể cùng

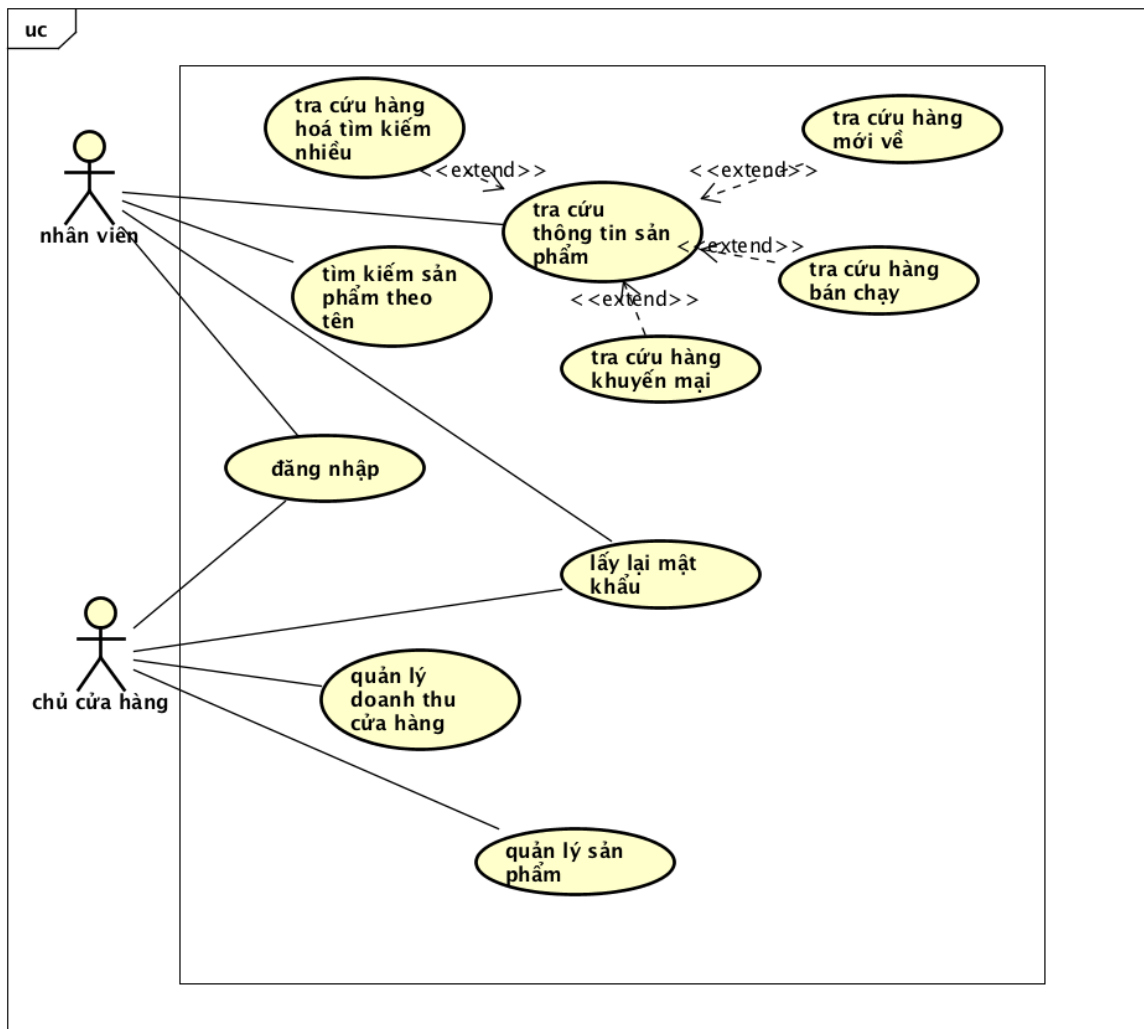
lúc quán xuyên hết được thông tin tại các cửa hàng đó, cũng như cập nhật xu hướng thời trang của khách hàng.

- Đối với nhân viên bán hàng việc nắm bắt chính xác thông tin về các sản phẩm trong cửa hàng là rất cần thiết để tư vấn cho khách hàng. Doanh thu của một cửa hàng phụ thuộc rất lớn vào khả năng bán hàng của mỗi nhân viên. Việc đưa ra thông tin tư vấn sẽ ảnh hưởng lớn đến quyết định sau đó của khách hàng. Ngay cả những nhân viên bán hàng lâu năm cũng không thể nắm bắt hết được danh mục các sản phẩm, tình trạng hiện tại của chúng.

2.2 Tổng quan chức năng

2.2.1 Biểu đồ use case tổng quan

Hình 1 dưới đây mô tả usecase tổng quan cho toàn bộ ứng dụng.



Hình 1 Biểu đồ usecase tổng quan

Trong ứng dụng có 2 tác nhân sử dụng chính gồm: nhân viên cửa hàng và chủ cửa hàng.

- Nhân viên cửa hàng: các nhân viên tại mỗi cửa hàng sẽ được cấp 1 tài khoản dùng để truy cập vào ứng dụng tư vấn. Tại ứng dụng này, nhân viên có thể tra cứu nhanh thông tin về sản phẩm của cửa hàng để tư vấn cho khách hàng của mình.
- Chủ cửa hàng: mỗi chủ cửa hàng cũng được cấp 1 tài khoản riêng để sử dụng ứng dụng quản lý của mình. Chủ cửa hàng có thể xem tình hình doanh thu của cả cửa hàng hoặc của từng chi nhánh theo thời gian. Ngoài ra chủ cửa hàng còn có thể xem được top doanh thu của các cửa hàng, top sản phẩm bán

chạy và top tìm kiếm của người dùng. Từ những thông tin đó, chủ cửa hàng có thể đưa ra các phân tích để có chiến lược kinh doanh hiệu quả

Các usecase chính bao gồm:

- Usecase đăng nhập: cho phép người dùng đăng nhập và sử dụng ứng dụng
- Usecase tra cứu thông tin sản phẩm: cho phép các nhân viên cửa hàng tìm kiếm nhanh thông tin sản phẩm theo size, màu sắc, ...
- Usecase tra cứu hàng mới về là một usecase mở rộng của usecase tra cứu thông tin sản phẩm. Usecase này cho phép hiển thị các sản phẩm mới nhất
- Usecase tra cứu hàng bán chạy cho phép hiển thị tất cả các sản phẩm đang bán chạy nhất của cửa hàng đó.
- Usecase tra cứu hàng khuyến mại cho phép hiển thị danh sách sản phẩm khuyến mại của cửa hàng.
- Usecase tra cứu hàng hoá tìm kiếm nhiều cho phép hiển thị danh sách các sản phẩm được khách hàng quan tâm, tìm kiếm nhiều nhất.
- Usecase tìm kiếm sản phẩm: cho phép nhân viên tìm kiếm nhanh thông tin về sản phẩm qua tên.
- Usecase quản lý doanh thu: cho phép chủ cửa hàng xem chi tiết doanh thu của từng cửa hàng theo thời gian
- Usecase quản lý sản phẩm: cho phép chủ cửa hàng xem top doanh thu, top mặt hàng bán chạy cũng như top tìm kiếm của người dùng.

2.3 Đặc tả Usecase chức năng

2.3.1 Đặc tả use case đăng nhập

Mã usecase	01	Tên usecase	Đăng nhập
Tác nhân	Nhân viên cửa hàng, chủ cửa hàng		

Tiền điều kiện	Không			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Hệ thống	Hiển thị giao diện đăng nhập	
	2.	Người 8ang	Nhập thông tin đăng nhập	
	3.	Hệ thống	Kiểm tra các trường dữ liệu nhập	
	4.	Hệ thống	Gửi thông tin đăng nhập lên server	
	5.	Hệ thống	Trả lại token đăng nhập	
	6.	Hệ thống	Lưu thông tin đăng nhập và token, đồng thời chuyển sang màn hình chính	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	3a	Hệ thống	Thông báo trường dữ liệu sai	
	5a	Hệ thống	Thông báo sai thông tin đăng nhập	
Hậu điều kiện	Không			

Hình 2 Đặc tả usecase đăng nhập

2.3.2 Đặc tả use case tìm kiếm sản phẩm theo tên

Mã usecase	02	Tên usecase	Tìm kiếm sản phẩm theo tên
Tác nhân	Nhân viên cửa hàng		

Tiền điều kiện	Đăng nhập với tài khoản nhân viên			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Nhân viên	Nhập tên tìm kiếm	
	2.	Hệ thống	Kiểm tra trường dữ liệu nhập có rỗng hay không	
	3.	Hệ thống	Gửi yêu cầu lên server	
	4.	Hệ thống	Nhận dữ liệu từ server và hiển thị	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	2a	Hệ thống	Thông báo tên tìm kiếm không được để trống	
Hậu điều kiện	Không			

Hình 3 Đặc tả usecase tìm kiếm sản phẩm theo tên

3.3.6 Đặc tả use case tra cứu hàng mới về

Mã usecase	03	Tên usecase	Tra cứu 9ang mới về	
Tác nhân	Nhân viên cửa hàng			
Tiền điều kiện	Đăng nhập với tài khoản nhân viên			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Nhân viên	Chọn option tra cứu 9ang mới về	

	2.	Hệ thống	Gửi yêu cầu lên server	
	3.	Hệ thống	Nhận dữ liệu trả về và hiển thị	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được	
Hậu điều kiện	Không			

Hình 4 Đặc tả use case tra cứu hàng mới về

2.3.5 Đặc tả use case tra cứu hàng bán chạy

Mã usecase	04	Tên usecase	Tra cứu 10ang bán chạy	
Tác nhân	Nhân viên cửa hàng			
Tiền điều kiện	Đăng nhập với tài khoản nhân viên			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Nhân viên	Chọn option tra cứu 10ang bán chạy	
	2.	Hệ thống	Gửi yêu cầu lên server	
	3.	Hệ thống	Nhận dữ liệu trả về và hiển thị	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	

	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được	
Hậu điều kiện	Không			

Hình 5 Đặc tả use case tra cứu hàng bán chạy

2.3.6 Đặc tả use case tra cứu hàng khuyến mại

Mã usecase	05	Tên usecase	Tra cứu hàng khuyến mại	
Tác nhân	Nhân viên cửa hàng			
Tiền điều kiện	Đăng nhập với tài khoản nhân viên			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Nhân viên	Chọn option tra cứu 11ang khuyến mại	
	2.	Hệ thống	Gửi yêu cầu lên server	
	3.	Hệ thống	Nhận dữ liệu trả về và hiển thị	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được	
Hậu điều kiện	Không			

2.3.7 Đặc tả use case tra cứu hàng hoá tìm kiếm nhiều

Mã usecase	06	Tên usecase	Tra cứu hàng hoá tìm kiếm nhiều
Tác nhân	Nhân viên cửa hàng		
Tiền điều kiện	Đăng nhập với tài khoản nhân viên		
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động
	1.	Nhân viên	Chọn option tra cứu hàng hoá tìm kiếm nhiều
	2.	Hệ thống	Gửi yêu cầu lên server
	3.	Hệ thống	Nhận dữ liệu trả về và hiển thị
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được
Hậu điều kiện	Không		

Hình 7 Đặc tả use case tra cứu hàng hoá tìm kiếm nhiều

2.3.8 Đặc tả use case quản lý doanh thu cửa hàng

Mã usecase	07	Tên usecase	Quản lý doanh thu cửa hàng
Tác nhân	Chủ cửa hàng		
Tiền điều kiện	Đăng nhập với tài khoản chủ cửa hàng		
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động
	1.	Nhân viên	Chọn chức năng quản lý doanh thu cửa hàng
	2.	Hệ thống	Gửi yêu cầu lên server
	3.	Hệ thống	Nhận dữ liệu trả về và vẽ đồ thị
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được
Hậu điều kiện	Không		

Hình 8 Đặc tả use case quản lý doanh thu cửa hàng

2.3.9 Đặc tả use case quản lý sản phẩm

Mã usecase	08	Tên usecase	Quản lý sản phẩm
Tác nhân	Chủ cửa hàng		

Tiền điều kiện	Đăng nhập với tài khoản chủ cửa hàng			
Luồng sự kiện chính	STT	Thực hiện bởi	Hành động	
	1.	Nhân viên	Chọn chức năng quản lý sản phẩm	
	2.	Hệ thống	Gửi yêu cầu lên server	
	3.	Hệ thống	Nhận dữ liệu trả về và vẽ đồ thị	
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động	
	2a	Hệ thống	Thông báo mất kết nối hoặc không kết nối được	
Hậu điều kiện	Không			

Hình 9 Đặc tả use case quản lý sản phẩm

2.4 Yêu cầu phi chức năng

Ứng dụng được sử dụng bởi cả nhân viên và chủ cửa hàng, do đó phải đảm bảo tính dễ dùng. Ngoài ra ứng dụng phải đáp ứng được những yêu cầu về bảo mật cơ bản trong việc gửi nhận dữ liệu.

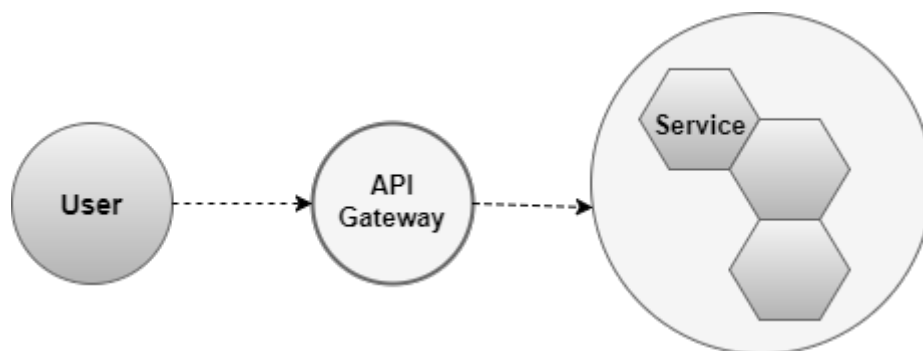
Ứng dụng tra cứu phục vụ cho nhân viên bán hàng nên tốc độ truy vấn dữ liệu phải nhanh hơn để đáp ứng yêu cầu phục vụ.

Chương 3 Công nghệ sử dụng

3.1 Microservice

Microservice là một mô hình hóa và thực thi hóa cho mô-hình-hướng-dịch-vụ¹, được sử dụng để xây dựng các hệ thống phần mềm linh hoạt và dễ triển khai.

Hình 1 là cấu trúc đơn giản nhất của một kiến trúc microservice. Tùy vào từng hệ thống, hướng phát triển mà Kiến trúc microservice có ít nhiều thay đổi. Nhưng nguyên lý hoạt động thì gần như không thay đổi.



Hình 10 Kiến trúc microservice

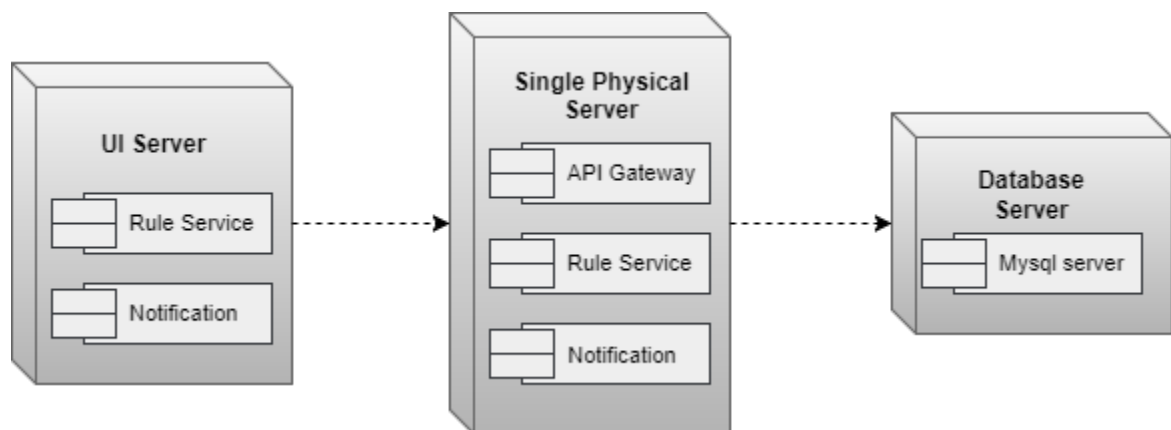
Hệ thống được chia nhỏ thành các Service, mỗi service có thể giao tiếp với một Cơ sở dữ liệu khác nhau. Người dùng giao tiếp với các service thông qua API Gateway. Khi bất cứ một service nào ngừng hoạt động do một nguyên nhân nào đó thì các service khác có thể hoạt động mà không bị ảnh hưởng.

Ưu điểm của kiến trúc microservice:

¹ Mô-hình-hướng-dịch-vụ: service-oriented architectures (SOA)

- Hiệu năng tốt
- Tái sử dụng được
- Dễ phát triển
- Dễ bảo trì
- Bảo mật tốt
- Khả năng chịu lỗi
- Khả năng tích hợp, mở rộng

Hình 2 dưới đây là một ví dụ cho mô hình triển khai hệ thống theo kiến trúc microservice.



Hình 11 Mô hình triển khai hệ thống ở mức cao

API Gateway: là một component² của mô hình Microservice. *API Gateway* là “lối vào duy nhất” cho tất cả các client. *API Gateway* xử lý các yêu cầu (request) bằng một trong hai cách, với các yêu cầu đơn giản chỉ là được định tuyến hoặc được ủy quyền cho service nhất định thì nó sẽ gửi đến các dịch vụ (service) phù hợp. Với các request khác nó xử lý bằng cách đưa ra nhiều dịch vụ (multiple services).

Notification: là một component của hệ thống. Notification là một microservice, nhiệm vụ của nó là gửi các thông báo như SMS, EMAIL, PUSH NOTIFICATION, v.v. Notification cũng quản lý mẫu cho nội dung của thông báo.

² Component: chỉ một thành phần cấu thành nên một đối tượng nào đó.

Registry: Registry là một "dịch vụ đăng ký"³, là một component của kiến trúc microservice, nó có thể được coi là cơ sở dữ liệu của các dịch vụ khác. Các cá thể dịch vụ (Service instances) được đăng ký các quyền, chức năng bởi service registry khi hệ thống khởi động, và được dỡ đăng ký khi hệ thống bị tắt. Khách hàng (client) của một Service nào đó truy vấn service registry để tìm kiếm đối tượng có sẵn của dịch vụ. Service registry có thể gọi một API để xác nhận xem một service có thể xử lý các yêu cầu không.

3.2 Nodejs

Tổng quan về Nodejs:

NodeJS là một mã nguồn được xây dựng dựa trên nền tảng Javascript V8 Engine, nó được sử dụng để xây dựng các ứng dụng web như các trang video clip, các forum và đặc biệt là trang mạng xã hội phạm vi hẹp. NodeJS là một mã nguồn mở được sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ Window cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

Các ưu điểm nổi bật của nodejs

- Bởi lẽ REST/JSON APIs gọn nhẹ là điều khiến NodeJS tỏa sáng. Với cơ chế event-driven, non-blocking I/O(Input/Output) và mô hình kết hợp với Javascript là sự lựa chọn tuyệt vời cho các dịch vụ Webs làm bằng JSON.
- NodeJS sẽ tận dụng tối đa Unix để hoạt động. Tức là NodeJS có thể xử lý hàng nghìn Process và trả ra 1 luồng khiến cho hiệu suất hoạt động đạt mức tối đa nhất và tuyệt vời nhất.

³ dịch vụ đăng ký : Service Registry

- Các web thông thường gửi HTTP request và nhận phản hồi lại (Luồng dữ liệu). Giả sử sẽ cần xử lý 1 luồng dữ liệu cực lớn, NodeJS sẽ xây dựng các Proxy phân vùng các luồng dữ liệu để đảm bảo tối đa hoạt động cho các luồng dữ liệu khác.
- Giả sử bạn xây dựng 1 ứng dụng chat, feed ... Facebook, Twitter là điển hình cho Web thực. NodeJS làm khá tốt điều đó!

3.3 Swift

Swift là ngôn ngữ chính thức được Apple thừa nhận để viết các sản phẩm mobile hoặc PC chạy trên hệ điều hành IOS. So với những ngôn ngữ khác, Swift có tuổi đời còn khá trẻ song không thể vì thế mà phủ nhận những ưu điểm vượt trội của ngôn ngữ này. Sau đây là một số đặc điểm của ngôn ngữ Swift mà em sử dụng để viết ứng dụng này:

- Swift là một ngôn ngữ lập trình hướng đối tượng. Sự ra đời của Swift có thể giúp các lập trình viên dễ dàng tiếp cận bởi nó sử dụng mã nguồn mở.
- Các dòng code của Swift ngắn và dễ đọc... giúp đẩy nhanh quá trình phát triển ứng dụng và tăng hiệu quả cho quá trình bảo trì trong tương lai.
- Swift tạo ra các ứng dụng có sức ép phần cứng thấp hơn Objective-C, người dùng sẽ được trải nghiệm game ấn tượng hơn trên các thiết bị iOS.
- Swift là một ngôn ngữ lập trình di động an toàn hơn so với người tiền nhiệm của nó

3.4 MySQL

MySQL⁴ là hệ quản trị cơ sở dữ liệu SQL nguồn mở phổ biến nhất, được phát triển, phân phối và hỗ trợ bởi Oracle Corporation.

MySQL là một hệ thống quản lý cơ sở dữ liệu.

Cơ sở dữ liệu là tập hợp có cấu trúc của dữ liệu. Nó có thể là bất cứ thứ gì từ một danh sách mua sắm đơn giản đến một thư viện hình ảnh hoặc một số lượng lớn thông tin trong một mạng lưới công ty. Để thêm, truy cập và xử lý dữ liệu được lưu trữ trong cơ sở dữ liệu, bạn cần một hệ thống quản lý cơ sở dữ liệu như MySQL Server. Vì các máy tính rất giỏi xử lý lượng lớn dữ liệu, các hệ thống quản lý cơ sở dữ liệu đóng vai trò trung tâm trong việc tính toán, như các tiện ích độc lập, hoặc như các phần của các ứng dụng khác.

MySQL là Cơ sở dữ liệu quan hệ.

Một cơ sở dữ liệu quan hệ lưu trữ dữ liệu trong các bảng riêng biệt thay vì đặt tất cả dữ liệu vào một kho lưu trữ lớn. Các cấu trúc cơ sở dữ liệu được tổ chức thành các tệp vật lý được tối ưu hóa cho tốc độ. Mô hình logic, với các đối tượng như cơ sở dữ liệu, bảng biểu, khung nhìn, hàng và cột, cung cấp môi trường lập trình linh hoạt. Bạn thiết lập các quy tắc điều chỉnh mối quan hệ giữa các trường dữ liệu khác nhau, chẳng hạn như một-một, một-nhiều, duy nhất, bắt buộc hoặc tùy chọn, và “các con trỏ” giữa các bảng khác nhau. Cơ sở dữ liệu thực thi các quy tắc này, do đó, với một cơ sở dữ liệu được thiết kế tốt, ứng dụng của bạn không bao giờ thấy dữ liệu không nhất quán, trùng lặp, đơn độc, đã lỗi thời hoặc thiếu.

Phần SQL của “MySQL” là viết tắt của “Structured Query Language”. SQL là ngôn ngữ tiêu chuẩn phổ biến nhất được sử dụng để truy cập cơ sở dữ liệu. Tùy thuộc vào môi trường lập trình của bạn, bạn có thể nhập SQL trực tiếp (ví dụ: để tạo báo cáo),

⁴ <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

nhúng các câu lệnh SQL vào mã được viết bằng ngôn ngữ khác hoặc sử dụng API cụ thể cho ngôn ngữ để ẩn cú pháp SQL.

Phần mềm MySQL là mã nguồn mở.

Mã nguồn mở có nghĩa là mọi người có thể sử dụng và sửa đổi phần mềm. Bất kỳ ai cũng có thể tải về phần mềm MySQL từ Internet và sử dụng nó mà không phải trả bất cứ điều gì. Nếu bạn muốn, bạn có thể nghiên cứu mã nguồn và thay đổi nó cho phù hợp với nhu cầu của bạn. Phần mềm MySQL sử dụng GPL (GNU General Public License), để xác định những gì bạn có thể và không thể làm với phần mềm trong các tình huống khác nhau. Nếu bạn cảm thấy không thoải mái với GPL hoặc cần phải nhúng mã MySQL vào một ứng dụng thương mại, bạn có thể mua một phiên bản thương mại được cấp phép.

Máy chủ cơ sở dữ liệu MySQL rất nhanh, đáng tin cậy, khả năng mở rộng và dễ sử dụng.

MySQL Server có thể chạy thoải mái trên máy tính để bàn hoặc máy tính xách tay, cùng với các ứng dụng khác của bạn, máy chủ web, v.v ... cần ít hoặc không có sự chú ý. Nếu bạn dành toàn bộ một máy cho MySQL, bạn có thể điều chỉnh các thiết lập để tận dụng tất cả bộ nhớ, sức mạnh của CPU và dung lượng I/O. MySQL cũng có thể quy mô lên đến các cụm máy móc, kết nối với nhau.

MySQL Server ban đầu được phát triển để xử lý cơ sở dữ liệu lớn nhanh hơn nhiều so với các giải pháp hiện tại và đã được sử dụng thành công trong các môi trường sản xuất đòi hỏi khắt khe trong nhiều năm. Mặc dù dưới sự phát triển liên tục, MySQL Server hôm nay cung cấp một bộ các chức năng phong phú và hữu ích. Khả năng kết nối, tốc độ và bảo mật của nó làm cho MySQL Server rất phù hợp với việc truy cập cơ sở dữ liệu trên Internet.

MySQL Server làm việc trong máy khách/máy chủ hoặc các hệ thống nhúng.

Phần mềm Cơ sở dữ liệu MySQL là một hệ thống máy khách/máy chủ bao gồm một máy chủ SQL đa luồng hỗ trợ các kết thúc ngược lại, một số chương trình khách hàng

khác nhau và thư viện, các công cụ hành chính và một loạt các giao diện lập trình ứng dụng (API).

Chúng tôi cũng cung cấp MySQL Server như một thư viện đa luồng nhúng mà bạn có thể liên kết với ứng dụng của bạn để có được sản phẩm độc lập nhỏ hơn, nhanh hơn và dễ quản lý hơn.

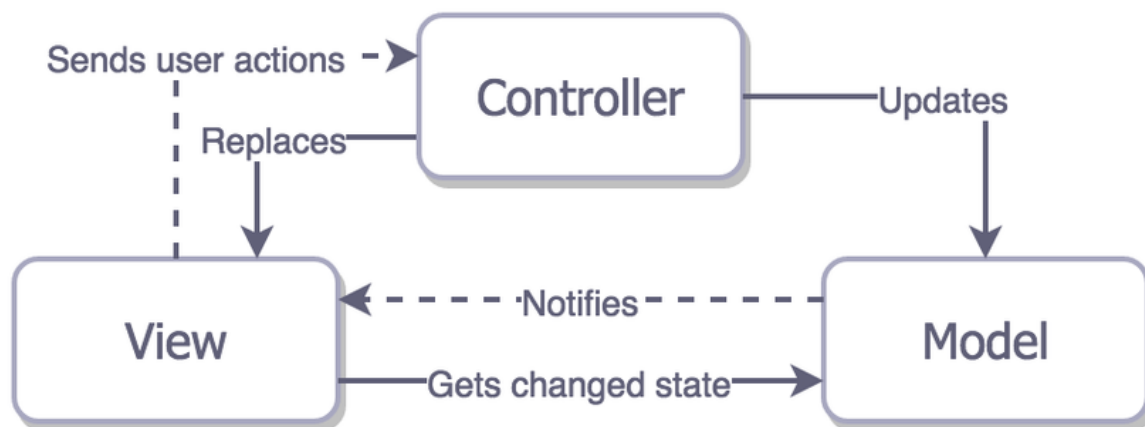
MySQL Server có một bộ tính năng thiết thực được phát triển với sự hợp tác chặt chẽ với người dùng của chúng tôi. Rất có thể ứng dụng hoặc ngôn ngữ yêu thích của bạn hỗ trợ máy chủ cơ sở dữ liệu MySQL.

Chương 4 Phát triển và triển khai ứng dụng

4.1 Thiết kế kiến trúc

4.1.1 Lựa chọn kiến trúc phần mềm

Trong đề tài ứng dụng của mình, em đã lựa chọn mô hình MVVM để xây dựng phía mobile. Với các ứng dụng mobile trước đây, chúng ta đã khá quen thuộc với mô hình kinh điển MVC (Module-View-Controller). Hình dưới đây mô tả cách thức hoạt động của mô hình MVC.

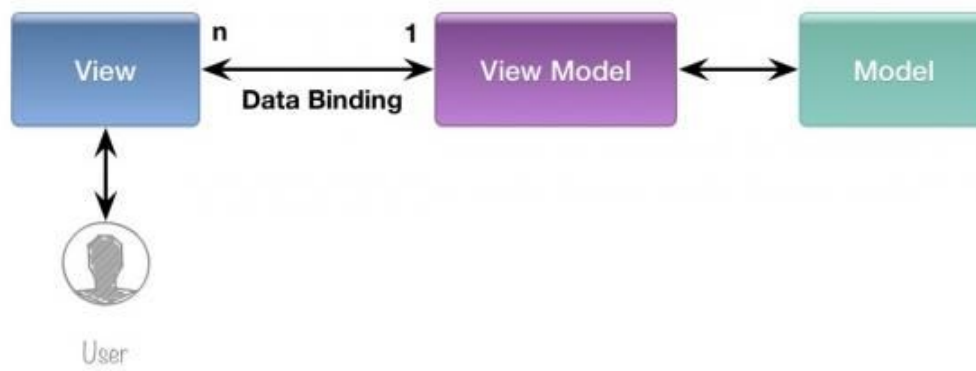


Hình 12 Mô hình hoạt động MVC

Đây là mô hình có từ lâu và rất được phổ biến trong quy trình phát triển phần mềm. Tuy nhiên một trong số những nhược điểm lớn nhất của mô hình khi áp dụng để triển khai ứng dụng mobile đó là việc phân tách nhiệm vụ giữa View và Controller

không được rõ ràng. Do còn tồn tại sự nhập nhằng giữa View và Controller dẫn đến việc test chức năng cho các ứng dụng trở lên khó khăn hơn.

MVVM là một mô hình được sinh ra nhằm khắc phục những khuyết điểm tồn tại của mô hình MVC. Hình dưới đây mô tả cách thức hoạt động của mô hình MVVM



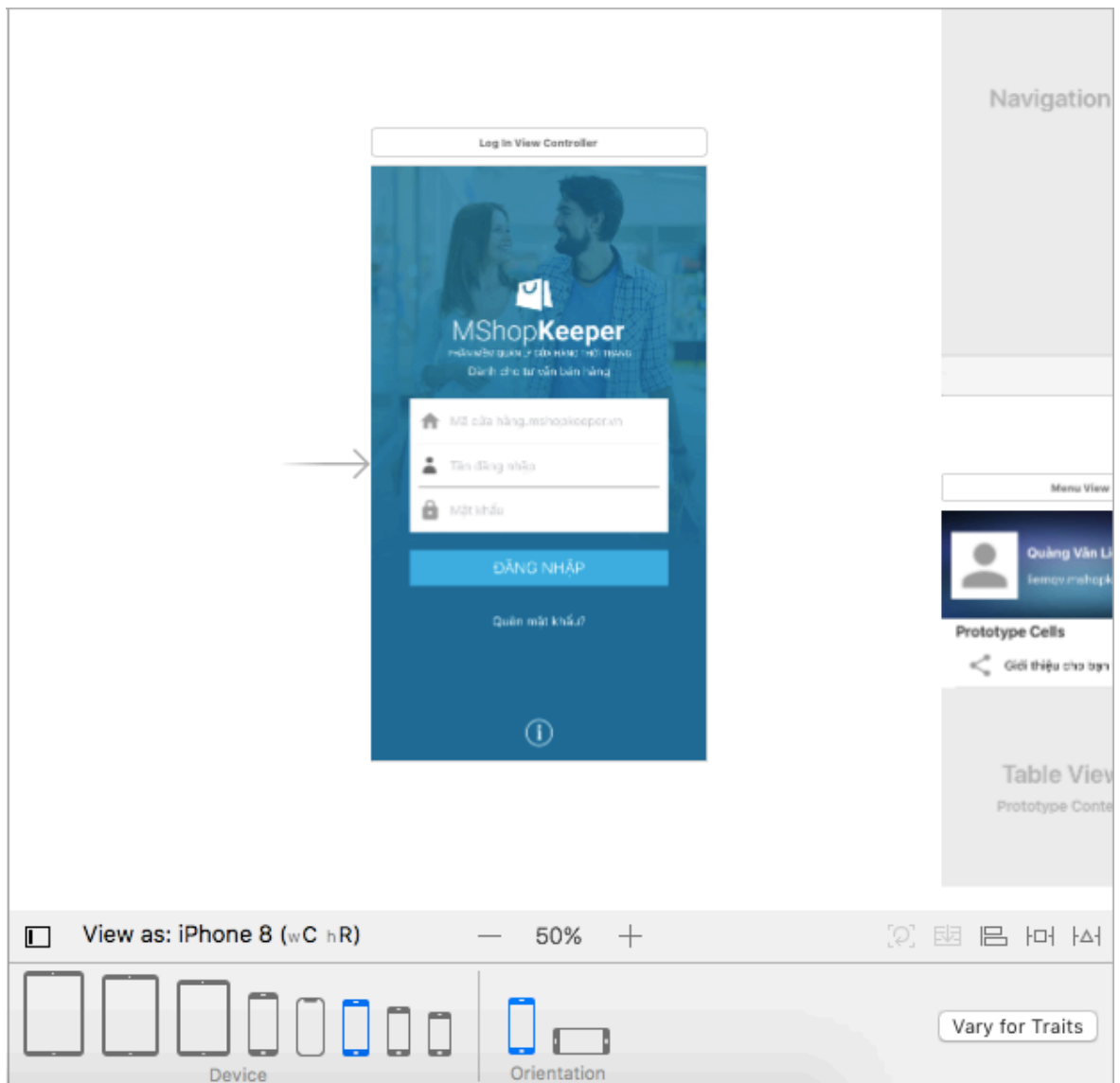
Hình 13 Mô hình MVVM

Thành phần ViewModel được sinh ra nhằm giúp điều hướng giữa View và Model. Từ đó giúp cho View không bị phình to khi project phát triển.

ViewModel có tính chất tái sử dụng trong code, giúp phân tách rõ ràng viewController và view. Trong ngôn ngữ Swift, thành phần View bao gồm các file storyboard, file xib, các file viewController

Ví dụ về mô hình MVVM được sử dụng trong ĐATN

Màn hình LoginViewController được thiết kế trong file MainStoryboard và đóng vai trò là View trong kiến trúc



Hình 14 View Login

Tiếp đến là file Account đóng vai trò Model (hình 15)

```

1 //
2 // User.swift
3 // MShopKeeper
4 //
5 // Created by ddthanh on 2/22/18.
6 // Copyright © 2018 ddthanh. All rights reserved.
7 //
8
9 import Foundation
10
11 class Account {
12 // khai báo các thuộc tính của đối tượng account
13 var username: String = ""
14 var password: String = ""
15 var userId: Int = 0
16 var fullName: String = ""
17 var tel: String = ""
18 var email: String = ""
19 var shop: [Shop]? = [Shop]()
20 var avatar: String = ""
21 }
22

```

Hình 15 File Account.swift

Cuối cùng là thành phần ViewModel trung gian điều khiển việc lưu dữ liệu và hiển thị. Trong file LoginViewController sử dụng 2 ViewModel là UserModel dùng để gọi phương thức lưu lại thông tin đăng nhập người dùng và ViewModel RequestAPI chứa các phương thức tương tác với API – hình

```

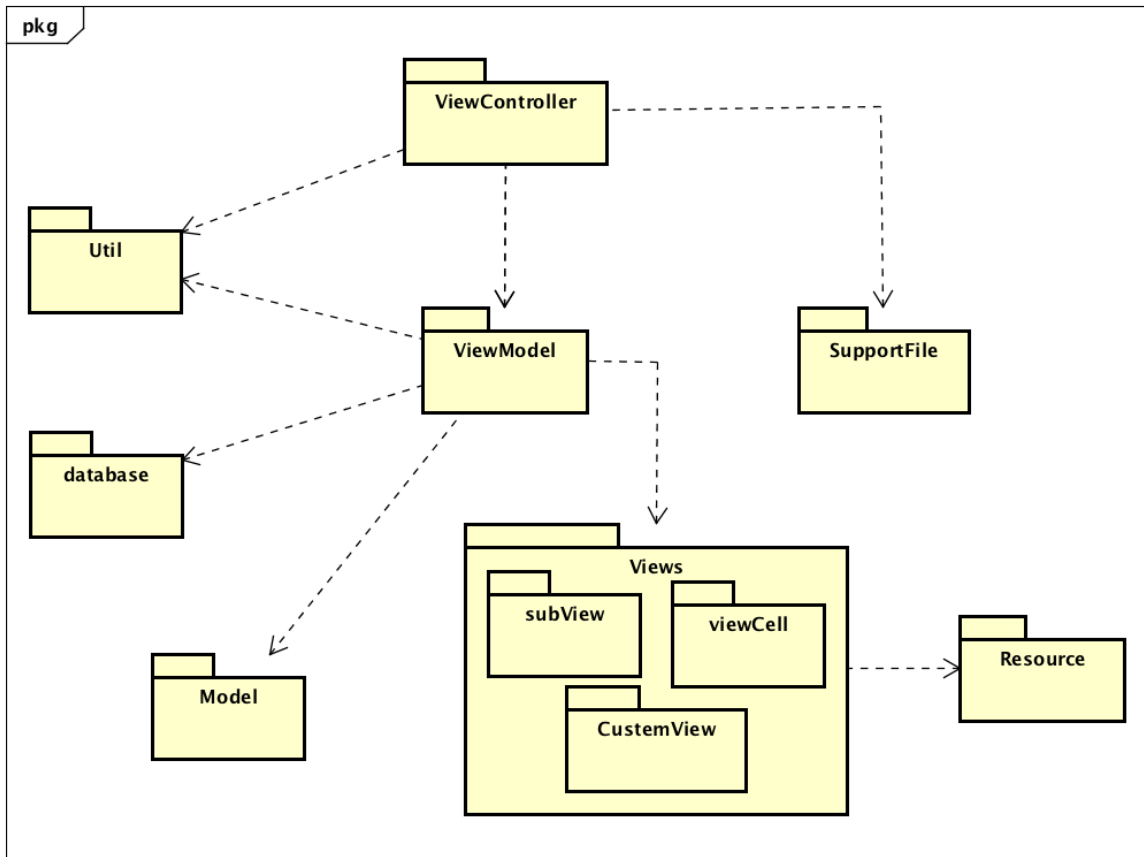
5 // các biến view model
6 var userModel: UserModel!
7 var requestAPI: RequestAPIModel!

```

Hình 16 ViewModel trong Login

Khi người dùng nhấn vào button đăng nhập view login sẽ gọi đến ViewModel RequestAPI thực thi phương thức login. Sau khi thực hiện xong, RequestAPI sẽ trả lại một số thông tin thông qua callback của hàm để view có thể tiếp tục thực hiện công việc của mình. Các luồng thực hiện logic sẽ không được viết trực tiếp trong các lớp View.

4.1.2 Thiết kế tổng quan



Hình 27 Ví dụ biểu đồ phụ thuộc gói

Mô tả chức năng các package:

- Package ViewController: Bao gồm các file Swift để quản lý các màn hình giao diện.
- Package Model: chứa các mô tả thực thể, đối tượng trong project.
- Package ViewModel: Gồm các file điều khiển việc hiển thị thành phần cho view.
- Package Util: Gồm các file định nghĩa các hàm tiện ích để các lớp trong dự án có thể gọi sử dụng.
- Package Resource: Gồm các file ảnh được sử dụng
- Package SupportFile: bao gồm các file quản lý của Xcode, file giao diện tổng quan.

- Package database: bao gồm các file khởi tạo và thực thi liên quan đến data local.

4.2 Thiết kế chi tiết

4.2.1 Thiết kế giao diện

Giao diện client được thiết kế để chạy tốt nhất trên các thiết bị iphone 6, 7, 8 với các độ phân giải màn hình như 750 x 1336 pixels và 1080 x 1920 pixels. Màn hình hiển thị với kích thước 4.7 inch và 5.5 inch cho phép hiển thị đầy đủ màu sắc.

Về các chuẩn thống nhất được mô tả như bảng sau:

Bảng 1 Quy chuẩn thiết kế giao diện ứng dụng

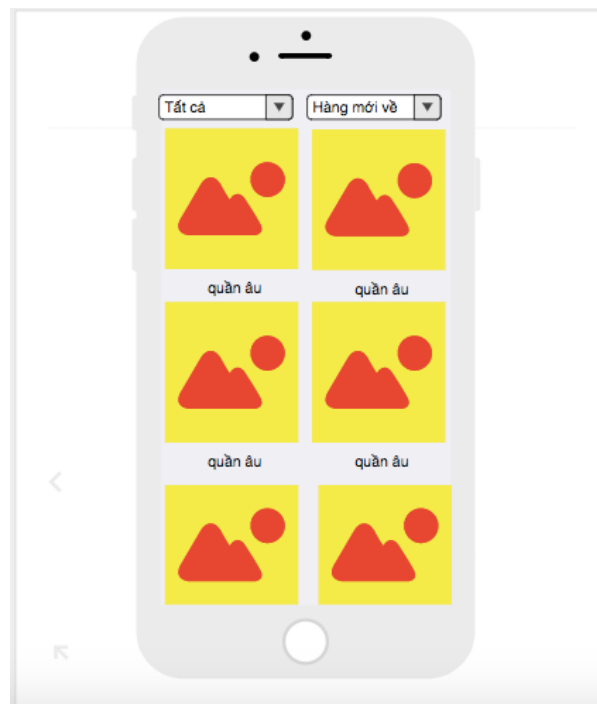
STT	Tên thành phần	Mô tả
1	Màu sắc	Ứng chọn chọn tông màu xanh với mã màu #006799 làm màu chủ đạo của ứng dụng, thống nhất màu sắc hiển thị giữa các màn hình
2	Phông chữ	Quy định phông chữ 14 cho các nội dung dạng text bình thường Phông chữ 17 cho các tiêu đề Phông chữ 12 cho các phần chú thích
3	Vị trí hiển thị thông báo	Đối với các thông báo, ứng dụng dùng stype chuẩn của Apple, hiển thị ở vị trí chính giữa màn hình thiết bị

4	Một số quy định thiết kế khác	Đối với các nút bấm trên màn hình phải được bo tròn và quy định chung một loại phông chữ hiển thị cũng như hiệu ứng khi tương tác
---	-------------------------------	---

Một số hình ảnh thiết kế minh hoạ cho chức năng của sản phẩm



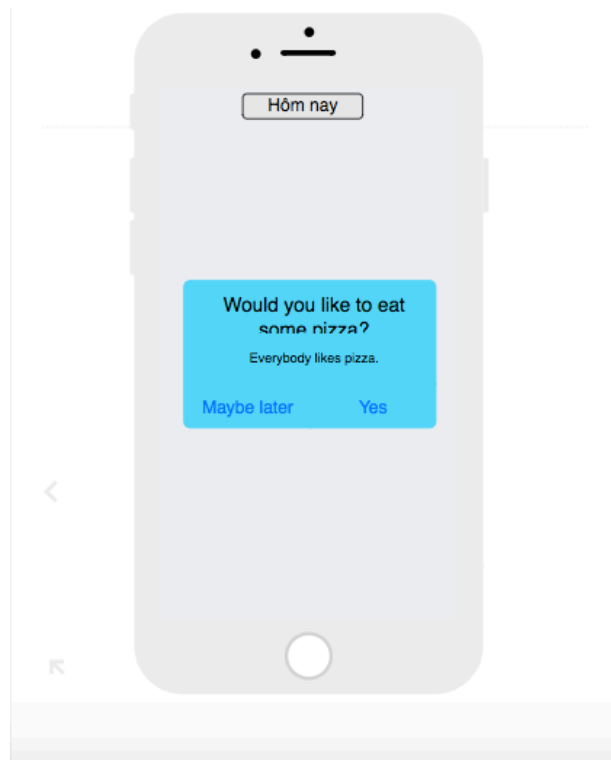
Hình 18 Minh hoạ các thành phần của Login



Hình 19 Minh họa cho chức năng tra cứu



Hình 20 Minh họa cho chức năng quản lý doanh thu



Hình 21 Minh hoạ thống nhất cách hiển thị thông báo

4.2.2 Thiết kế lớp

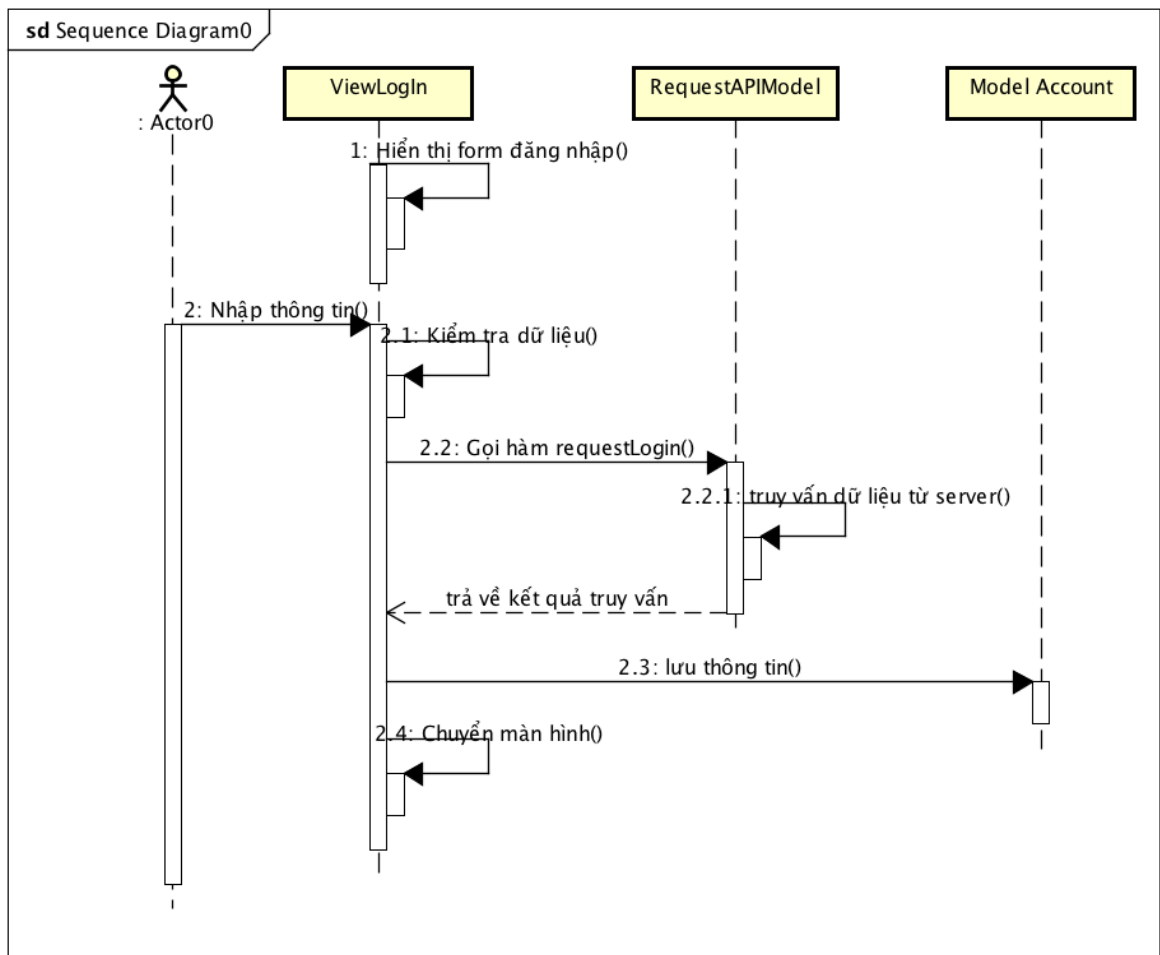
Thiết kế chi tiết cho một số lớp quan trọng của project.

Class RequestAPIModel: class này dùng để định nghĩa các hàm tương tác với WebAPI. Trong class này sẽ mô tả các phương thức truy vấn dữ liệu qua một số hàm như requestLogin(), getAllModel(), forgetPass()...

Class Model_SQLite: định nghĩa các phương thức tương tác với dữ liệu local. Trong lớp này có các hàm cụ thể như init(), insert(), delete() hay getModelForKey().

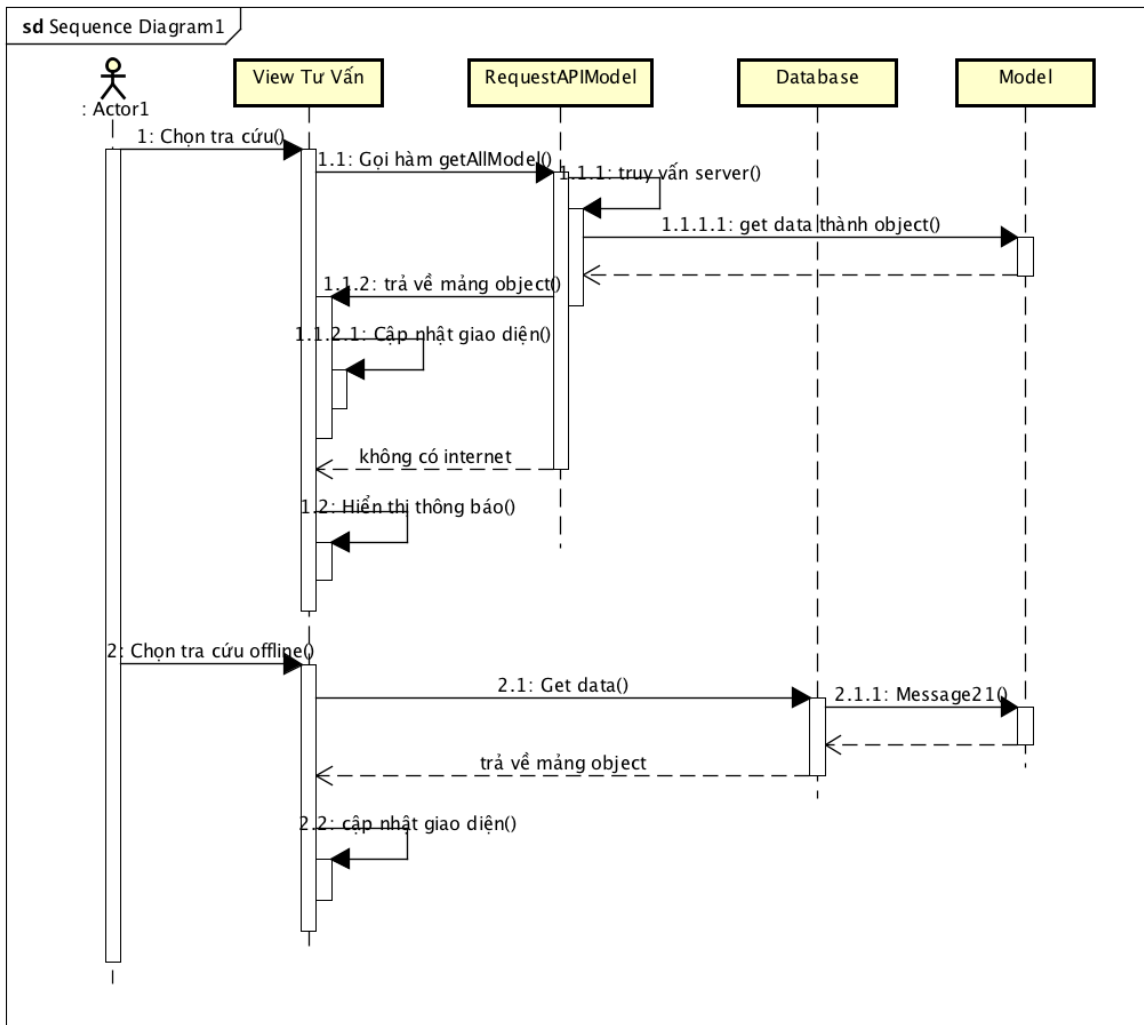
Ngoài ra còn một số class liên quan đến việc hiển thị dữ liệu người dùng, các class chứa phương thức tiện ích nhằm thực hiện cho một số vấn đề nhỏ khác.

Minh hoạ vai trò của các class thông qua các biểu đồ tuần tự sau.



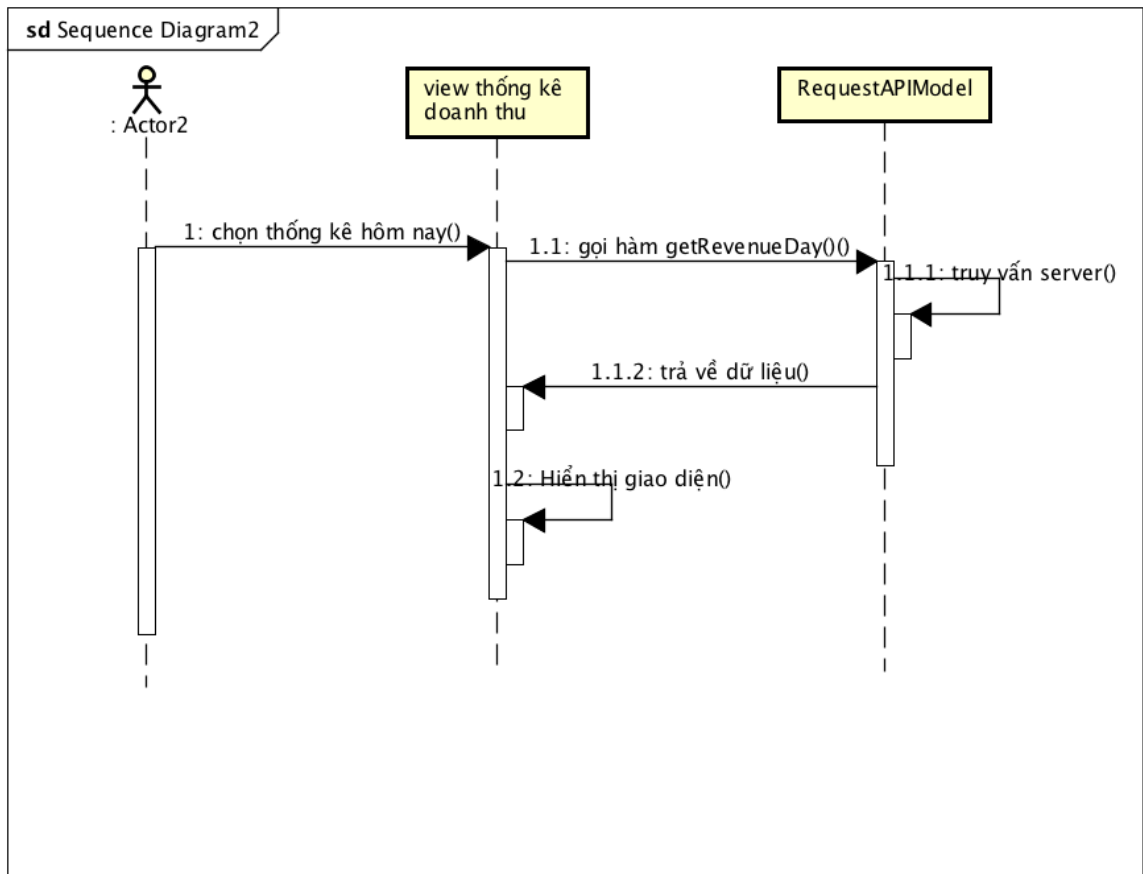
Hình 22 Biểu đồ tuần tự cho chức năng đăng nhập

Hình 23 mô tả chức năng hiển thị danh sách các model trong cửa hàng. Tùy vào lựa chọn của nhân viên tra cứu theo tiêu chí gì ứng dụng sẽ hiển thị danh sách tương ứng.



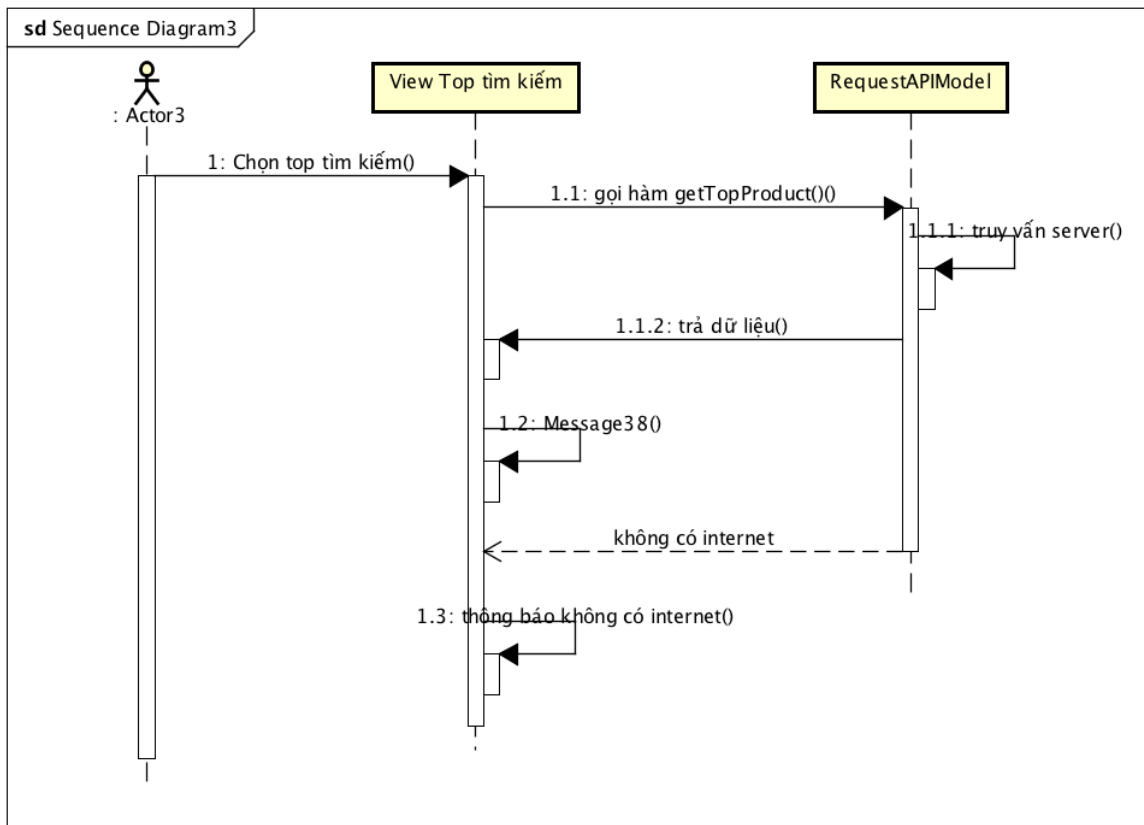
Hình 23 Biểu đồ trình tự cho chức năng tra cứu theo tên

Hình 24 mô tả trình tự cho chức năng thống kê doanh thu của cửa hàng. Tại màn hình thống kê, chủ cửa sẽ có nhiều lựa chọn thống kê theo ngày, tuần hay tháng. Ngoài ra ứng dụng còn hỗ trợ cho phép hiển thị thống kê doanh thu theo từng chi nhánh trong toàn bộ cửa hàng. Với chức năng này sẽ giúp chủ cửa hàng có những đánh giá chính xác nhất về tình hình kinh doanh của mỗi chi nhánh, qua đó sẽ có những chiến lược kinh doanh và nhập hàng phù hợp.



Hình 24 Biểu đồ tuần tự thống kê doanh thu

Hình 25 mô tả trình tự cho chức năng hiển thị top sản phẩm tìm kiếm nhiều nhất tại cửa hàng. Nắm bắt được xu hướng tìm kiếm sản phẩm của khách hàng là một trong những yếu tố cần thiết nhất để chủ cửa hàng có thể đưa ra được những chiến lược kinh doanh hợp lý. Từ những con số thống kê cho từng cửa hàng, chủ cửa hàng cũng dễ dàng hơn trong việc cung cấp các mẫu sản phẩm cho từng cửa hàng đó. Không chỉ có thế, chỉ số thống kê còn là nguồn tham khảo để chủ cửa hàng nắm bắt được nhu cầu của khách hàng, xu hướng thời trang theo mùa để kịp thời bổ sung vào những thời gian sau.



Hình 25 Biểu đồ tuần tự cho chức năng hiển thị top tìm kiếm sản phẩm

4.2.3 Thiết kế cơ sở dữ liệu

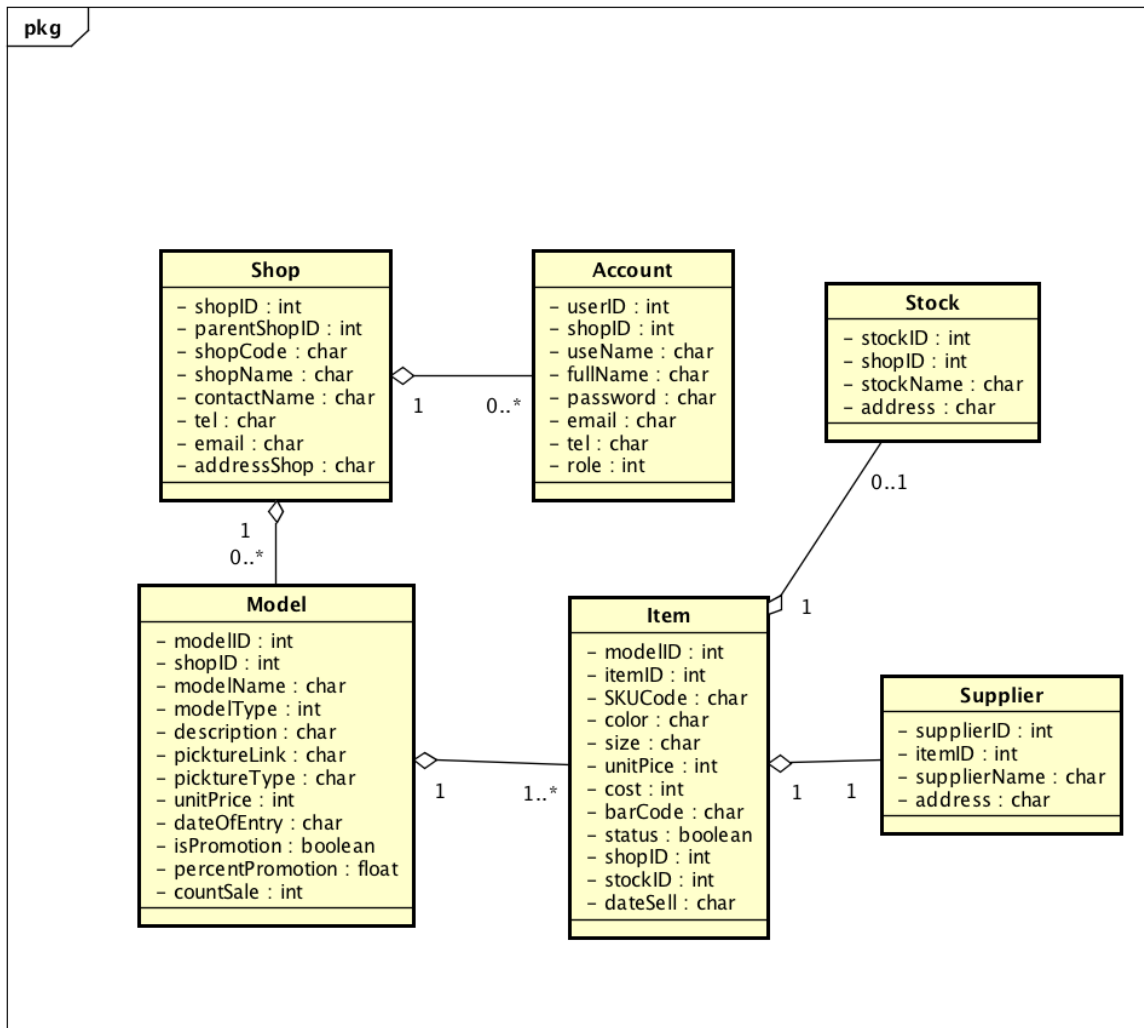
Phần này có độ dài từ hai đến bốn trang. Sinh viên thiết kế, vẽ và giải thích biểu đồ thực thể liên kết (E-R diagram). Từ đó, sinh viên thiết kế cơ sở dữ liệu tùy theo hệ quản trị cơ sở dữ liệu mà mình sử dụng (SQL, NoSQL, Firebase, v.v.)

Phân tích bài toán tra cứu tư vấn và quản lý doanh thu cửa hàng như sau:



Hình 26 các thực thể liên kết

Hình 27 dưới đây mô tả các thuộc tính của thực thể và mối quan hệ liên kết giữa các thực thể trong bài toán.



Hình 27 Mô hình thực thể liên kết

Hình 28 mô tả thiết kế các bảng cơ sở dữ liệu

mshopkeeper_database shop shopID : int(8) # parentShopID : int(8) shopCode : varchar(128) shopName : varchar(128) contactName : varchar(128) tel : varchar(16) email : varchar(128) addressShop : varchar(128)	mshopkeeper_database model modelID : int(8) # shopID : int(8) modelName : varchar(128) # modeType : int(1) decription : varchar(128) pictureLink : varchar(128) # pictureType : int(1) unitPrice : int(8) # isDisplayPrice : int(1) dateOfEntry : date # isPromotion : int(1) percenPromotion : float saleCount : int(16)	mshopkeeper_database account userID : int(8) # shopID : int(8) userName : varchar(128) fullName : varchar(128) password : char(8) email : varchar(128) tel : varchar(16) # role : int(1)
mshopkeeper_database item itemID : int(8) # modelID : int(8) SKUCode : char(8) decription : varchar(128) color : varchar(128) size : varchar(128) unitPrice : int(16) cost : int(11) # isDisplayPrice : int(1) dateOfEntry : date # isPromotion : int(1) percenPromotion : float barCode : varchar(32) # shopID : int(8) # stockID : int(8) # status : int(11) dateSell : date	mshopkeeper_database stock stockID : int(8) # shopID : int(8) stockName : varchar(128) address : varchar(128)	mshopkeeper_database supplier supplierID : int(8) # itemID : int(8) supplierName : varchar(128) address : varchar(128)

Hình 28 Các bảng cơ sở dữ liệu

4.3 Xây dựng ứng dụng

4.3.1 Thư viện và công cụ sử dụng

Trong quá trình thực hiện đề tài của mình, em có tham khảo và sử dụng một số công cụ lập trình và các thư viện hỗ trợ như sau:

Bảng 2 Danh sách thư viện và công cụ sử dụng

Mục đích	Công cụ	Địa chỉ URL
IDE lập trình	Xcode	http://www.eclipse.org/
	Atom	https://atom.io/
Thư viện	Alamofire	https://github.com/Alamofire/Alamofire

Kingfisher	https://github.com/onevc/Kingfisher
SlideMenuSwift	https://github.com/dekatotoro/SlideMenuControllerSwift
Express	https://expressjs.com/
JWT	https://github.com/auth0/node-jsonwebtoken

4.3.2 Kết quả đạt được

Sản phẩm sau khi hoàn thành bao gồm các thành phần sau:

- File .zip của project Mshopkeeper tư vấn: đây là file nén của ứng dụng tư vấn dành cho nhân viên cửa hàng.
- File .zip của project Mshopkeeper thống kê: đây là file nén của ứng dụng quản lý dành cho chủ cửa hàng.
- File .zip của project server và API: đây là file nén toàn bộ project của phần server

Thống kê khác

Stt	Tên thành phần	Kích thước
1	Tổng số dòng code	
2	Số lớp	
3	Số gói	
4	Dung lượng app mshopkeeper tư vấn	
5	Dung lượng app mshopkeeper quản lý	
6	Dung lượng project server	

4.3.3 Minh họa các chức năng chính

Sinh viên lựa chọn và đưa ra màn hình cho các chức năng chính, quan trọng, và thú vị nhất. Mỗi giao diện cần phải có lời giải thích ngắn gọn. Khi giải thích, sinh viên có thể kết hợp với các chú thích ở trong hình ảnh giao diện.

4.4 Kiểm thử

Phần này có độ dài từ hai đến ba trang. Sinh viên thiết kế các trường hợp kiểm thử cho hai đến ba chức năng quan trọng nhất. Sinh viên cần chỉ rõ các kỹ thuật kiểm thử đã sử dụng. Chi tiết các trường hợp kiểm thử khác, nếu muốn trình bày, sinh viên đưa vào phần phụ lục.

Sinh viên sau cùng tổng kết về số lượng các trường hợp kiểm thử và kết quả kiểm thử. Sinh viên cần phân tích lý do nếu kết quả kiểm thử không đạt.

4.5 Triển khai

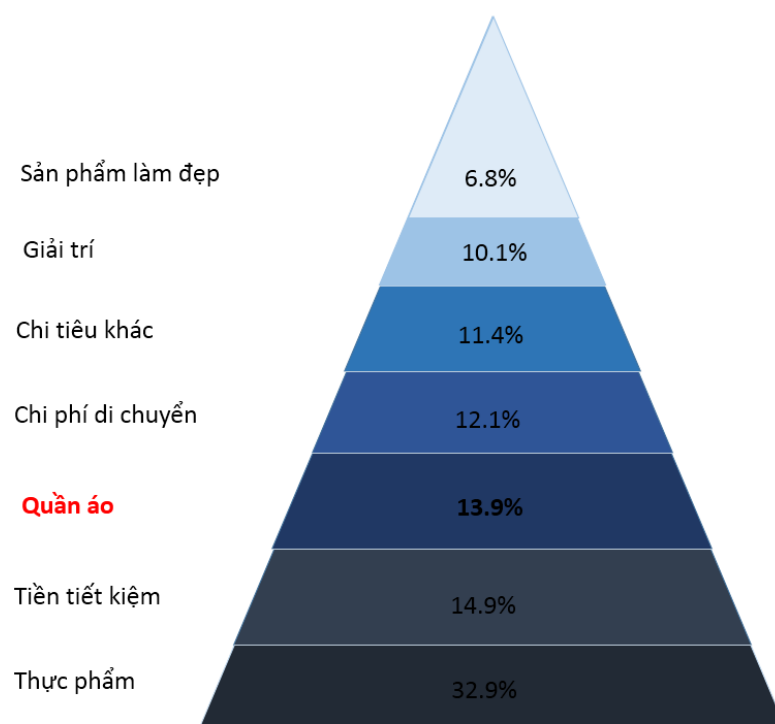
Sinh viên trình bày mô hình và/hoặc cách thức triển khai thử nghiệm/thực tế. Ứng dụng của sinh viên được triển khai trên server/thiết bị gì, cấu hình như thế nào. Kết quả triển khai thử nghiệm nếu có (số lượng người dùng, số lượng truy cập, thời gian phản hồi, phản hồi người dùng, khả năng chịu tải, các thống kê, v.v.)

Chương 5 Các giải pháp và đóng góp nổi bật

5.1 Những đóng góp của đề tài trong thực tế.

Với yêu cầu bài toán đặt ra không quá mới mẻ, xong đề tài thực sự đã đem đến những tiện ích thiết thực đối với người dùng sản phẩm.

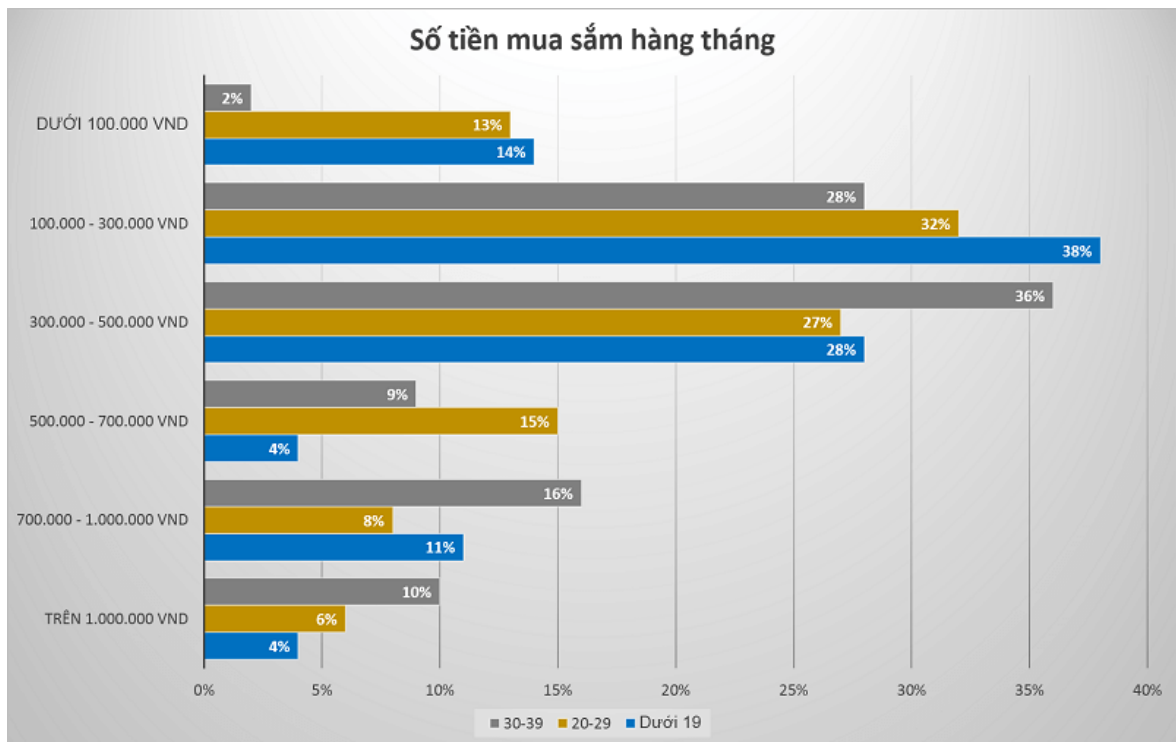
Phân tích nhu cầu thời trang thực tế của người tiêu dùng Việt Nam trong một vài năm trở lại đây để thấy được thị trường này đang có tiềm năng phát triển đến mức nào. Theo một bài báo thống kê cuối năm 2017 đã đưa ra những phân tích đánh giá chi tiết về nhu cầu cũng như thói quen của người dùng. Hình dưới đây cho thấy phân trăm chi phí mà người dùng dành cho mua sắm thời trang trong 1 tháng.



Hình 29:

Theo số liệu điều tra, mức chi tiêu dành cho quần áo của người Việt đứng thứ 3 chiếm 13,9% chỉ sau chi tiêu dành cho thực phẩm (32,9%) và tiền tiết kiệm (14,9%). Đây là một con số khá cao so với chi tiêu và tiết kiệm của người tiêu dùng Việt.

Trung bình người tiêu dùng Việt sẽ chi tiêu trong khoảng 1 Triệu đồng cho việc mua sắm quần áo trong một tháng. Những sản phẩm giá thành thấp, trong khoảng 100 – 300 ngàn được lựa chọn nhiều nhất. Theo sau đó là mức giá từ 300 – 500 ngàn. Khách hàng có độ tuổi từ 30 – 39 là khách hàng mua nhiều nhất.



Hình 30 Số tiền mua sắm hàng tháng của người dùng.

Từ số liệu trên ta có thể dễ dàng nhận thấy những mặt hàng có giá vừa phải luôn là sự lựa chọn hàng đầu cho giới trẻ. Đó cũng là lý do tại sao hiện nay có nhiều cá nhân, cửa hàng xuất hiện để nhắm mục tiêu đến các sản phẩm dành cho người trẻ. Những cửa hàng, cá nhân này có chung một đặc điểm là hình thức kinh doanh ở mức độ trung bình do hạn chế về chi phí, hình thức kinh doanh cũng khá là đa dạng, từ những cửa hàng, ki-ốt nhỏ hay thậm chí kinh doanh online. Càng ngày số lượng đối tượng lựa chọn hình thức kinh doanh này càng lớn. Đi kèm với đó cũng là nhu cầu về một phần mềm hỗ trợ tư vấn và quản lý cho người buôn bán. Không giống như những cửa hàng thời trang lớn, người buôn bán theo hình thức này không thể tự bỏ tiền để xây dựng cho mình những ứng dụng riêng. Chính vì thế sản phẩm mà đề tài ĐATN của em hướng đến sẽ là một giải pháp hoàn toàn phù hợp và mang lại những lợi ích cụ thể.

Ngoài những đáp ứng thiết yếu và phù hợp cho đối tượng kinh doanh thời trang vừa và nhỏ, sản phẩm này cũng là một sự lựa chọn phù hợp đối với những cửa hàng thời trang lớn. Sản phẩm MShopkeeper có những chức năng hoàn toàn phù hợp với những yêu cầu thực tế của các cửa hàng thời trang hiện nay. Một ưu điểm khá quan trọng đó

là sản phẩm có tập trung và thực hiện chức năng phân tích số liệu quản lý khá chi tiết. Các chủ shop thời trang cửa hàng có thể dựa vào những con số này để đưa ra những kế hoạch kinh doanh cụ thể cho cửa hàng của mình.

5.2 Những khó khăn và giải pháp đặt ra trong ĐATN

Trong quá trình làm ĐATN của mình, em đã gặp một vài những khó khăn như sau:

5.2.1 Giải quyết vấn đề tối ưu truy vấn.

i. Dẫn dắt/giới thiệu về bài toán/vấn đề

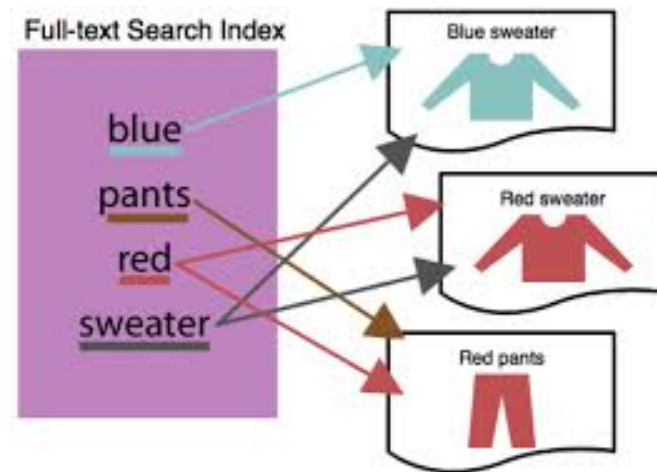
Trong quá trình thiết kế và xây dựng cơ sở dữ liệu cho ứng dụng, một trong những vấn đề quan trọng nhất là đảm bảo tốc độ truy vấn dữ liệu. Cơ sở dữ liệu của ứng dụng phải đáp ứng cho nhiều cửa hàng khác nhau, mỗi cửa hàng lại có rất nhiều sản phẩm khác nhau. Chính vì thế số lượng bản ghi trong cơ sở dữ liệu là rất lớn và ngày càng được mở rộng hơn. Khi số lượng cửa hàng đăng ký tăng lên cũng là lúc bài toán tối ưu truy vấn được đặt ra. Một ứng dụng có phải là sự lựa chọn tốt nhất cho người dùng không thì ngoài việc phải đáp ứng những chức năng thiết yếu còn phải đáp ứng hiệu năng và trải nghiệm người dùng. Sẽ thật là tồi tệ nếu như bất người dùng phải chờ đợi. Theo số liệu thống kê hiện nay, thời gian trung bình người dùng có thể chấp nhận được đối với 1 thao tác là không quá 3s. Nếu không xử lý tốt vấn đề truy vấn thì 1 request dữ liệu từ cơ sở mà em đã thiết kế sẽ không thể đáp ứng được nhu cầu của khách hàng.

ii. Giải pháp

Từ bản chất của ứng dụng chủ yếu là truy vấn và lấy dữ liệu từ server trả về để hiển thị theo mục đích của người dùng do đó em lựa chọn giải pháp đặt index cho một số trường trong cơ sở dữ liệu.

Index trong các cơ sở dữ liệu thực chất là cũng là 1 loại bảng mà giữ khoá

chính hoặc trường index và một loại con trỏ đến từng bản ghi vào 1 bảng thực tế. Có thể coi index giống như một bảng tra cứu phụ lục trong các cuốn sách. Người sử dụng không thể nhìn thấy các index, chúng chỉ được sử dụng để tăng tốc độ truy vấn và sẽ được sử dụng bởi cơ sở dữ liệu tìm kiếm để xác định vị trí bản ghi rất nhanh. Hình dưới đây mô tả khái quát chức năng của index trong các cơ sở dữ liệu.



Hình 31 Mô tả chức năng của index trong cơ sở dữ liệu

5.2.2 giải quyết vấn đề bảo mật

i. Dẫn dắt/giới thiệu về bài toán/vấn đề

Bảo mật là một trong những yêu cầu quan trọng đối với các ứng dụng di động nói riêng và các sản phẩm phần mềm nói chung. Các dữ liệu có thể coi như một tài sản riêng của từng cửa hàng, từng cá nhân. Vì thế việc bảo mật dữ liệu là điều cần thiết. Một ứng dụng tốt bắt buộc phải đáp ứng những yêu cầu về bảo mật dữ liệu người dùng. Trong đề tài của mình, vấn đề bảo mật cũng được em chú ý đến khi thiết kế và xây dựng chương trình.

ii. Giải pháp

Do sử dụng phương thức truyền nhận dữ liệu từ server tới client qua các API nên em đã lựa chọn phương thức bảo mật sử dụng thư viện JSON WEB TOKEN (JWT). JWT được biết đến rộng rãi như một giải pháp hỗ trợ bảo mật tốt nhất cho API. Cùng tìm hiểu qua về cách thức hoạt động của JWT để

đúng thông tin yêu cầu thông qua hàm verify().

Bước 6: Server gửi lại thông tin cho client.

Việc gửi nhận yêu cầu dữ liệu thông qua xác thực token sẽ giúp đảm bảo an toàn hơn cho dữ liệu người dùng, tránh tối đa việc thất thoát trong quá trình truyền tin. Ngoài ra việc gửi token kèm theo header sẽ giảm số câu truy vấn lặp lại trong database, từ đó nâng cao được thời gian xử lý các yêu cầu.

Thông thường mỗi 1 token sẽ có một khoảng thời hạn nhất định. Trong ứng dụng của mình em đang cấu hình khoảng thời gian cho 1 phiên làm việc là 24h. Sau khi token hết hiệu lực, người dùng phải đăng nhập lại ứng dụng để bắt đầu một phiên làm việc mới.

Chương 6 Kết luận và hướng phát triển

6.1 Kết luận

Tổng kết sau quá trình thực hiện đề tài và đánh giá chung với một số ứng dụng cùng loại trên thị trường, em rút ra một số kết luận sau:

- Ứng dụng tra cứu tư vấn và quản lý cửa hàng Mshopkeeper đã đáp ứng tốt các yêu cầu đề ra của bài toán.
- Ứng dụng tuy không được tích hợp tính năng bán hàng như một vài sản phẩm cùng loại trên thị trường xong cũng phần nào mang lại lợi ích thực sự cho các hộ kinh doanh và cửa hàng.
- Ứng dụng có khả năng triển khai với nhiều tập người dùng, đáp ứng tốt các trải nghiệm cơ bản.
- Với mục tiêu và đối tượng ban đầu hướng đến là các hộ kinh doanh vừa và nhỏ, ứng dụng này thực sự có tiềm năng phát triển.
- Hiện tại ứng dụng còn một vài điểm hạn chế, chưa tích hợp được tính năng hỗ trợ bán hàng, đặt hàng. Ứng dụng còn phải cải tiến nhiều hơn về mặt hiệu năng nếu tập người dùng lớn.

Qua quá trình tìm hiểu và thực hiện đề tài em cũng phần nào rút ra cho mình những bài học từ việc áp dụng các lý thuyết cho đến quá trình tạo ra một sản phẩm thực tế hữu ích với người dùng.

6.2 Hướng phát triển

Trước tiên, sinh viên trình bày các công việc cần thiết để hoàn thiện các chức năng/nhiệm vụ đã làm. Sau đó sinh viên phân tích các hướng đi mới cho phép cải thiện và nâng cấp các chức năng/nhiệm vụ đã làm.

Để sản phẩm có thể hoàn thiện và tồn tại được trong tương lai, em đã có một số định hướng nhằm cải thiện và phát triển ứng dụng như sau.

Trước hết cần hoàn thiện hơn một số yêu cầu quản lý, tập trung thống kê chi tiết hơn thói quen của người dùng.

Đánh giá được các thông số khác như tỉ lệ mua hàng của người dùng theo từng giờ trong ngày, các khung giờ cao điểm.

Trong tương lai sản phẩm cần tích hợp và có thể mở rộng đối tượng sử dụng đến cả khách mua hàng chứ không chỉ dừng lại cho nhân viên và chủ cửa hàng. Khi đó cần thêm cả tính năng hỗ trợ tự động tư vấn dựa vào thông tin mà khách hàng nhập.

Tài liệu tham khảo

Lưu ý: Sinh viên không được đưa bài giảng/slide, các trang Wikipedia, hoặc các trang web thông thường làm tài liệu tham khảo.

Một trang web được phép dùng làm tài liệu tham khảo **chỉ khi** nó là công bố chính thống của cá nhân hoặc tổ chức nào đó. Ví dụ, trang web đặc tả ngôn ngữ XML của tổ chức W3C <https://www.w3.org/TR/2008/REC-xml-20081126/> là TLTK hợp lệ.

Có năm loại tài liệu tham khảo mà sinh viên phải tuân thủ đúng quy định về cách thức liệt kê thông tin như sau:

* **Bài báo đăng trên tạp chí khoa học:** Tên tác giả, tên bài báo, tên tạp chí, volume, từ trang đến trang (nếu có), nhà xuất bản, năm xuất bản.

[1] Hovy E. H., Automated Discourse Generation Using Discourse Structure Relations, Artificial Intelligence, Elsevier Science Publishers, 63: 341-385, 1993.

* **Sách:** Tên tác giả, tên sách, volume (nếu có), lần tái bản (nếu có), nhà xuất bản, năm xuất bản.

[2] Peterson L. L. and Davie B. S. , Computer Networks: A Systems Approach, 2nd ed., Morgan-Kaufmann, 1999.

[3] Nguyễn Thúc Hải, Mạng máy tính và các hệ thống mở, Nhà xuất bản giáo dục, 1999.

* **Tập san Báo cáo Hội nghị Khoa học:** Tên tác giả, tên báo cáo, tên hội nghị, ngày (nếu có), địa điểm hội nghị, năm xuất bản.

- [4] Poesio M. and Di Eugenio B., Discourse Structure and Anaphoric Accessibility, In Proc. of the ESSLLI Workshop on Information Structure, Discourse Structure and Discourse Semantics, Helsinki, 2001.

* **Đồ án tốt nghiệp, Luận văn Thạc sĩ, Tiến sĩ:** Tên tác giả, tên đồ án/luận văn, loại đồ án/luận văn, tên trường, địa điểm, năm xuất bản.

- [5] Knott D., A Data-Driven Methodology for Motivating a Set of Coherence Relations, Ph.D. Thesis, University of Edinburgh, UK, 1996.

* **Tài liệu tham khảo từ Internet :** Tên tác giả (nếu có), tựa đề, cơ quan (nếu có), địa chỉ trang web, thời gian lần cuối truy cập trang web.

- [6] Berners-Lee T., Hypertext Transfer Protocol (HTTP), CERN,
<ftp://info.cern.ch/pub/www/doc/http-spec.txt.Z>, last visited May 2010.

- [7] Princeton University, WordNet,
<http://www.cogsci.princeton.edu/~wn/index.shtml>, last visited May 2010.

A Hướng dẫn viết đề án tốt nghiệp

A.1 Quy định chung

Dưới đây là một số quy định và hướng dẫn viết đề án tốt nghiệp mà bắt buộc sinh viên phải **đọc kỹ và tuân thủ nghiêm ngặt**.

Sinh viên cần đảm bảo tính thống nhất toàn báo cáo (font chữ, căn dòng hai bên, hình ảnh, bảng, margin trang, đánh số trang, v.v.). Để làm được như vậy, sinh viên chỉ cần sử dụng các định dạng theo đúng template ĐATN này. Khi paste nội dung văn bản từ tài liệu khác của mình, sinh viên cần chọn kiểu Copy là “Text Only” để định dạng văn bản của template không bị phá vỡ/vi phạm.

Tuyệt đối cấm sinh viên đạo văn. Sinh viên cần ghi rõ nguồn cho tất cả những gì không tự mình viết/vẽ lên, bao gồm các câu trích dẫn, các hình ảnh, bảng biểu, v.v. Khi bị phát hiện, sinh viên sẽ **không được phép bảo vệ ĐATN**.

Tất cả các hình vẽ, bảng biểu, công thức, và tài liệu tham khảo trong ĐATN nhất thiết phải được SV giải thích và tham chiếu tới ít nhất một lần. Không chấp nhận các trường hợp sinh viên đưa ra hình ảnh, bảng biểu tùy hứng và không có lời mô tả/giải thích nào.

Sinh viên tuyệt đối không trình bày ĐATN theo kiểu viết ý hoặc gạch đầu dòng. ĐATN không phải là một slide thuyết trình; khi người đọc không hiểu sẽ không có ai giải thích hộ. Sinh viên cần viết thành các đoạn văn và phân tích, diễn giải đầy đủ, rõ ràng. Câu văn cần đúng ngữ pháp, đầy đủ chủ ngữ, vị ngữ và các thành phần câu.

Khi thực sự cần liệt kê, sinh viên nên liệt kê theo phong cách khoa học với các ký tự La Mã. Ví dụ, nhiều sinh viên luôn cảm thấy hối hận vì (i) chưa cố gắng hết mình, (ii) chưa sắp xếp thời gian học/choi một cách hợp lý, (iii) chưa tìm được người yêu để chia sẻ quãng đời sinh viên vất vả, và (iv) viết ĐATN một cách cầu thả.

Trong một số trường hợp nhất thiết phải dùng các bullet để liệt kê, sinh viên cần thống nhất Style cho toàn bộ các bullet các cấp mà mình sử dụng đến trong báo cáo. Nếu dùng bullet cấp 1 là hình tròn đen, toàn bộ báo cáo cần thống nhất cách dùng như vậy; ví dụ như sau:

- Đây là mục 1 – Thực sự không còn cách nào khác tôi mới dùng đến bullet trong báo cáo.
- Đây là mục 2 – Nghĩ lại thì tôi có thể không cần dùng bullet cũng được. Nên tôi sẽ xóa bullet và tổ chức lại hai mục này trong báo cáo của mình cho khoa học hơn. Tôi muốn thầy cô và người đọc cảm nhận được tâm huyết của tôi trong từng trang báo cáo ĐATN.

A.2 Tạo đề mục

Đề mục giúp tạo bố cục cho tài liệu. Để các tính năng tự động – ví dụ tính năng cập nhật mục lục, hoặc tính năng tham chiếu chéo – của Word hoạt động được, sinh viên cần tuân thủ theo các style đã tạo trong tài liệu này. Để hiển thị các style này, sinh viên vào tab Home trong thanh Ribbon của Word.

Để tạo đề mục cấp 1, 2, 3, 4, 5, sinh viên gõ tiêu đề cho đề mục của mình rồi chọn các style là Heading 1, 2, 3, 4, 5 tương ứng. Sinh viên hạn chế dùng tới đề mục cấp 4, và phải trong trường hợp thực sự cần thiết mới dùng đến đề mục cấp 5.

Phần phụ lục chỉ cho phép có hai cấp tiêu đề. Hai style tương ứng với hai cấp này là “Heading 7, Phụ lục cấp 1” và “Heading 8, Phụ lục cấp 2”.

A.3 Bảng biểu

Sinh viên lưu ý không để bảng tràn ra lề (margin) trên, dưới, trái hoặc phải của trang. Do không gian nhỏ hẹp, bảng nên có font là 12pt, nhỏ hơn một chút so với font thông thường (13pt) của ĐATN. Độ dẫn dòng của bảng nên là 1 line. Căn lề bảng là căn giữa, nhưng nội dung văn bản trong bảng nên được căn lề trái.

Sinh viên có thể viết tắt các từ trong bảng để tiết kiệm không gian nhưng phải giải thích các từ viết tắt này ở phần Chú thích bảng. Ví dụ áp dụng được minh họa trong Bảng 1.

Bảng 1 Ví dụ sử dụng bảng

Chú thích: Y: Year; RS: Risk Set; G: Graduated; AB: Academically Excluded; C: Censored; HRG: Hazard Ratio – Graduated

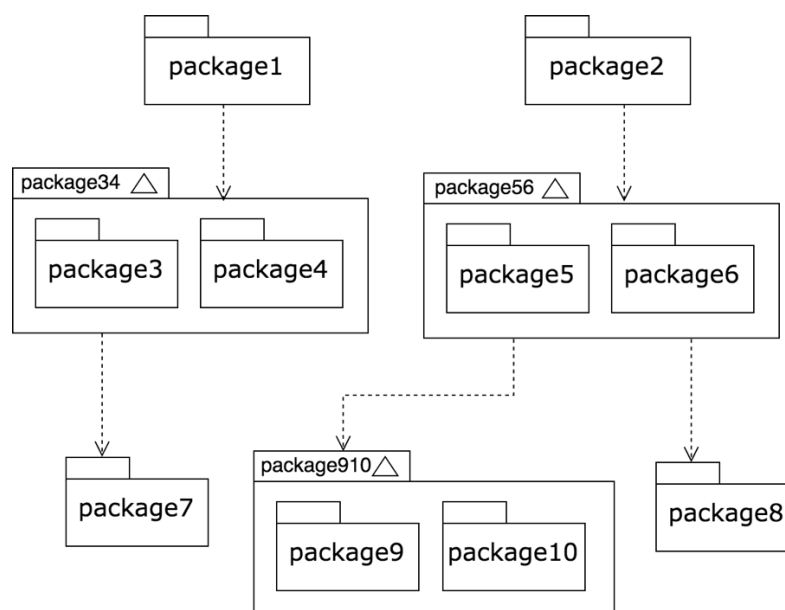
Y	RS	G	AE	C	HRG (%)
1	11.959	0	725	1619	0,0
2	10.457	0	474	1513	0,0
3	7.365	1213	335	966	16,7
4	900	599	145	405	55,3

Sinh viên được tùy ý lựa chọn định dạng (template) cho các bảng trong ĐATN của mình, nhưng phải đảm bảo tính nhất quán trên toàn ĐATN. Template bảng phải đảm bảo phần heading cột trong bảng có font in đậm và nổi bật (highlighted) hơn so với các nội dung khác trong bảng.

Vì bảng có thể kéo dài nhiều trang, tiêu đề của bảng nên để ở phần đầu của bảng. Sinh viên không thêm tiêu đề bảng bằng tay. Để thêm tiêu đề bảng tự động, sinh viên nhấn chuột phải vào bảng, chọn “Insert Caption”, chọn “Label” là “Bảng”, rồi nhấn nút “OK”. Sau đó, sinh viên nhập vào nội dung tiêu đề và căn chỉnh “Center” cho tiêu đề này. Lưu ý, sinh viên cần bôi đậm bằng tay cụm từ **Bảng n**. Kết quả thu được có dạng như sau “**Bảng 1** Ví dụ sử dụng bảng”. Sinh viên nên xoay ngang trang giấy với bảng có nhiều cột hoặc nhiều nội dung văn bản.

A.4 Hình vẽ

Tương tự như bảng, sinh viên không được để hình vẽ tràn lề trang. Căn lề cho hình vẽ là căn giữa (Center). Cách thêm tiêu đề hình vẽ tương tự như cách thêm tiêu đề bảng, nhưng sinh viên chọn “Label” là “Hình” thay vì “Bảng”.



Hình 3 Ví dụ hình vẽ

Tiêu đề hình vẽ phải đặt ở dưới hình vẽ. Nếu hình vẽ được copy từ trên mạng, sinh viên bắt buộc phải ghi rõ nguồn. Sinh viên nên thống nhất công cụ sử dụng và style cho hình vẽ trong toàn ĐATN. Các chi tiết trong hình vẽ phải được bố trí gọn gàng; chữ trong hình phải đảm bảo nhìn được rõ nét khi in báo cáo trên giấy A4. Khi resize ảnh, cần giữ nguyên tỷ lệ dài rộng, tránh làm méo hoặc vỡ hình. Ví dụ hình vẽ được minh họa trong Hình 3.

A.5 Tài liệu tham khảo

Sinh viên cần hạn chế tối đa dùng trang Web làm tài liệu tham khảo. Chỉ chấp nhận trang Web làm TLTK khi trang đó là nơi công bố chính thức của tổ chức hoặc cá nhân nào đó. Ví dụ, trang web đặc tả ngôn ngữ XML của tổ chức W3C <https://www.w3.org/TR/2008/REC-xml-20081126/> là TLTK hợp lệ.

Khi giới thiệu về công cụ, API, thư viện, hoặc nền tảng nào đó, sinh viên có thể đưa ra địa chỉ URL của các tiện ích này. Sinh viên lưu ý địa chỉ URL đó không phải là tài

liệu tham khảo. Trong các trường hợp tương tự như vậy, sinh viên nên tạo “Footnote”. Sinh viên tạo “Footnote” bằng cách vào mục “References”, chọn “Insert Footnote”. Ví dụ tạo Footnote như sau: TensorFlow⁵ là nền tảng học máy mã nguồn mở đang được sử dụng rộng rãi hiện nay. Lưu ý: số Footnote phải đặt sát với từ được mô tả. Như trong ví dụ trên, số 1 được đặt ngay cạnh chữ TensorFlow (không có dấu cách).

A.6 Công thức toán học

Giống như bảng, hình vẽ, và tài liệu tham khảo, công thức toán học cần được đánh số, giải thích, và tham chiếu đầy đủ.

Để thêm tiêu đề (caption) cho công thức, sinh viên đặt con trỏ văn bản vào dòng văn bản dưới công thức. Sau đó vào menu “Insert” (lưu ý là menu Insert, không phải là tab Insert trong thanh Ribbon) và chọn mục “Caption”. Từ Popup Menu hiện ra, sinh viên chọn “Label” là “Công thức”, rồi nhấn nút “OK”. Sinh viên bôi đậm chữ **Công thức n**, và căn tiêu đề ra giữa. Công thức 1 là ví dụ mẫu cho sinh viên tham khảo.

$$(x + a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$

Công thức 1 Khai triển Newton

A.7 Tham chiếu chéo

Tham chiếu chéo (Cross-reference) là tiện ích hữu hiệu cho người viết báo cáo. Nó giúp tạo các liên kết tham chiếu (hyperlink) tới các hình ảnh, bảng biểu, tài liệu tham khảo, và các đề mục một cách tự động. Ví dụ, ngay trong câu này, một tham chiếu đã được tạo ra tới mục 2.1. Người đọc dễ dàng nhấp chuột vào liên kết 2.1 để ngay lập tức chuyển đến mục đó.

⁵ <https://www.tensorflow.org/>, lần truy cập cuối: 28/06/2018

Để tạo tham chiếu chéo tới các đề mục (heading), sinh viên vào tab “References”, rồi tìm và nhấn chọn “Cross-reference”. SV chọn “Reference type” là “Heading” và chọn “Insert Reference to” là “Heading number (no context)”. Sau đó, SV chọn phần đề mục muốn tham chiếu rồi bấm “Insert”.

Để tạo tham chiếu chéo tới các hình vẽ, bảng biểu và công thức, sinh viên cũng vào tab “References” và chọn “Cross-reference”. SV chọn “Reference type” là “Hình”, “Bảng”, hoặc “Công thức”. SV chọn “Insert Reference to” là “Only label and number”, trở đến phần muốn tham chiếu rồi bấm “Insert”. Nếu font chữ trong liên kết tham chiếu tạo ra được in đậm (bold), SV chuyển về dạng thường cho chuẩn tắc.

Thực hiện tương tự các bước mô tả ở trên, sinh viên có thể tạo tham chiếu chéo tới các tài liệu tham khảo. Sinh viên chọn “Reference type” là “Numbered item”, chọn “Insert Reference to” là “Paragraph number”, trở đến phần tài liệu tham khảo muốn tham chiếu rồi bấm “Insert”. Ví dụ, tham chiếu chéo tới tài liệu tham khảo [2], [3], [4] đã được tạo.

A.8 Cập nhật mục lục và tham chiếu chéo

Trong suốt quá trình viết ĐATN, sinh viên sẽ tạo ra nhiều xáo trộn như thay đổi vị trí hình và bảng, thay đổi thứ tự tài liệu tham khảo, thay đổi tên đề mục, v.v. Vì vậy, để hoàn thành ĐATN, sinh viên cần cập nhật lại các thành phần mục lục, danh sách hình ảnh/bảng/công thức và tham chiếu chéo được tạo tự động trong ĐATN của mình.

Để cập nhật các thành phần này, sinh viên bấm Ctrl+A để chọn toàn báo cáo, nhấn chuột phải và chọn “Update Field”, rồi liên tục chọn mục “Entire table” khi được hỏi trong hộp thoại Popup Menu. Sau đó sinh viên search chữ “Error” trên toàn báo cáo để kiểm tra xem có lỗi đánh chỉ mục hoặc lỗi tham chiếu nào không. Đồng thời, sinh viên nên tự soát lại bằng tay toàn bộ nội dung quyển ĐATN để tránh mọi sai sót.

A.9 In quyển đồ án tốt nghiệp

Do hiện nay có nhiều phiên bản Word cho nhiều nền tảng máy tính, sinh viên nhất thiết phải xuất ĐATN ra định dạng PDF rồi mang tới cửa hàng in ấn để tránh sai sót.

Sinh viên chỉ cần in bìa giấy cứng mà không cần đóng quyển để tiết kiệm chi phí và vận chuyển ĐATN dễ dàng hơn.

Quyển ĐATN nên được in trên các trang giấy trắng đủ dày. Mỗi tờ giấy A4 chỉ được in một mặt.