

TRƯỜNG ĐẠI HỌC SƯ PHẠM ĐÀ NẴNG

KHOA TIN HỌC

————o0o————

BÁO CÁO KHÓA LUẬN TỐT NGHIỆP

**XÂY DỰNG HỆ THỐNG
TRA CỨU DINH DƯỠNG VÀ QUẢN LÝ THỰC ĐƠN
CHUYÊN NGÀNH: CỬ NHÂN CÔNG NGHỆ THÔNG TIN**

Giảng viên hướng dẫn : TS. Nguyễn Trần Quốc Vinh

Sinh viên thực hiện : Đồng Diễm Thúy

Lớp : 11CNTT1

ĐÀ NẴNG - 2015

LỜI CẢM ƠN

Đầu tiên, em xin bày tỏ lòng biết ơn chân thành và sâu sắc tới Tiến sĩ Nguyễn Trần Quốc Vinh, người đã tận tình hướng dẫn, giúp đỡ em để luận văn này được hoàn thành.

Em xin chân thành cảm ơn Trường Đại học Sư Phạm Đà Nẵng cùng toàn thể các giảng viên của khoa Tin học, trường Đại học Sư Phạm Đà Nẵng đã tạo mọi điều kiện thuận lợi cho em trong quá trình học tập và hoàn thành luận văn tốt nghiệp đại học. Em xin được cảm ơn sự dạy dỗ, chỉ bảo và quan tâm của các thầy cô khoa Tin học trong suốt thời gian em theo học và hoàn thành luận văn.

Cuối cùng, em muốn gửi lời cảm ơn chân thành tới gia đình, bạn bè, những người luôn động viên khích lệ giúp em hoàn thành luận văn. Xin chân thành cảm ơn.

Sinh viên: Đồng Diễm Thúy

Lớp: 11CNTT1

MỤC LỤC

DANH MỤC VIẾT TẮT	i
DANH MỤC BẢNG BIỂU	iii
DANH MỤC HÌNH VẼ	iv
MỞ ĐẦU	1
1 Tính cấp thiết của đề tài	1
2 Mục tiêu nghiên cứu	4
3 Đối tượng và phạm vi nghiên cứu	4
4 Phương pháp nghiên cứu	5
5 Bố cục đề tài	6
1 CƠ SỞ LÝ THUYẾT	7
1 Sơ đồ tổng thể của hệ thống	7
2 Giao thức làm việc của hệ thống	9
3 RESTful Web Service	12
4 Công nghệ JAX-RS	14
5 Servlet	16
6 Phương pháp phát triển ứng dụng dựa trên mô hình TDD (Test-Driven Development)	19

7	Phương pháp phát triển ứng dụng web dựa trên mô hình SPA	23
7.1	Giới thiệu về SPA	23
7.2	Ưu và nhược điểm của SPA	23
8	AngularJS	25
8.1	Các khái niệm dùng trong AngularJS (xem [1]) . . .	25
8.2	Mô hình MVC trong AngularJS	27
9	Xây dựng ứng dụng di động trên Ionic Framework	28
9.1	Giới thiệu về Ionic Framework	28
9.2	Cordova	29
9.3	Tính năng nổi bật của Ionic Framework	29

2 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 33

1	Khái quát nghiệp vụ bài toán	33
2	Hướng tiếp cận bài toán	35
3	Chức năng của hệ thống	38
4	Các tác nhân của hệ thống	38
5	Lược đồ Usecase của toàn hệ thống	39
5.1	Mô tả Usecase dưới hình thức User Story	39
5.2	Mô tả Acception Test	40
6	Sequence Diagram	43
7	Class Diagram	47
8	Cơ sở dữ liệu	49
9	Mô tả dữ liệu	50
10	Thiết kế giao diện	55

3	KẾT QUẢ ĐẠT ĐƯỢC	59
1	Các giao diện	59
1.1	Giao diện trang bìa	59
1.2	Giao diện nutrix_webservice	60
1.3	Giao diện nutrix_ngadmin	61
1.4	Giao diện nutrix_webapp	63
1.5	Giao diện nutrix_mobiapp	68
2	Cài đặt và cấu hình hệ thống	68
2.1	Cài đặt MySQL và Apache Server	69
2.2	Java	70
2.3	Import cơ sở dữ liệu vào MYSQL sử dụng phpMyAdmin	73
2.4	Tomcat Server và nutrix-service	75
3	Mã nguồn chương trình	76
	KẾT LUẬN VÀ KIẾN NGHỊ	77
	TÀI LIỆU THAM KHẢO	78
	PHỤ LỤC	83
1	Mã nguồn chương trình	84

DANH MỤC VIẾT TẮT

STT	Kí hiệu	Tên của ký hiệu viết tắt	Tên tiếng việt
1	FAO	Food and Agriculture Organization of the United Nations	Tổ chức Lương thực và Nông nghiệp Liên Hiệp Quốc
2	WHO	World Health Organization	Tổ chức Y tế Thế giới
3	USDA	United States Department of Agriculture	Bộ Nông nghiệp Hoa Kỳ
4	BMI	Body Mass Index	Chỉ số khối cơ thể
5	TDD	Test Driven Development	Mô hình phát triển với trọng tâm hướng về việc kiểm thử
6	API	Application Programming Interface	Giao diện lập trình ứng dụng
7	REST	Representational State Transfer Web Service	

8	SPA	Single Page Application	Ứng dụng trang duy nhất
9	HTTP	Hypertext Transfer Protocol	Giao thức truyền siêu văn bản
10	JDBC	Java Database Connectivity	Công nghệ kết nối cơ sở dữ liệu Java
11	URL	Uniform Resource Location	Định vị tài nguyên thống nhất
12	JSON	JavaScript Object Notation	
13	XML	eXtensible Markup Language	Ngôn ngữ đánh dấu mở rộng
14	DOM	Document Object Model	Mô hình đối tượng tài liệu
15	CSS	Cascading Style Sheet	

DANH MỤC BẢNG BIỂU

2.1	Tác nhân của hệ thống	38
2.2	Bảng mô tả Acceptance Test cho chức năng tìm kiếm thành phần dinh dưỡng trong thực phẩm	42
2.3	Cấu trúc thông tin dữ liệu của Ingredient	50
2.4	Cấu trúc thông tin dữ liệu của nutrient	51
2.5	Cấu trúc thông tin dữ liệu của bảng ingredient_join_nutrient	51
2.6	Cấu trúc thông tin dữ liệu của bảng nutrix_recipe	52
2.7	Cấu trúc thông tin dữ liệu bảng nutrix_recipe_join_ingredient	52
2.8	Bảng mô tả cấu trúc thông tin dữ liệu thành phần dinh dưỡng của thực phẩm thịt heo nạc (pork) của Hoa Kỳ.	55

DANH MỤC HÌNH VẼ

1.1	Sơ đồ tổng quan của hệ thống	7
1.2	Giao thức làm việc của hệ thống	9
1.3	Hoạt động của Servlet	16
1.4	Chu trình của Test Driven Development	20
1.5	Mô hình MVC trong AngularJS	27
2.1	Bài toán đặt ra của hệ thống	35
2.2	Sơ đồ mối quan hệ giữa các phương diện của hệ thống.	36
2.3	Các phương tiện hệ thống sử dụng.	38
2.4	Lược đồ usecase của hệ thống.	39
2.5	Sơ đồ tuần tự cho chức năng tìm kiếm thông tin của một loại thực phẩm trên giao diện	43
2.6	Sơ đồ tuần tự cho chức năng tìm kiếm thông tin loại thực phẩm khi người dùng cung cấp tên một loại thực phẩm cho service	43
2.7	Sơ đồ tuần tự cho chức năng tìm kiếm thông tin của một chất dinh dưỡng trên giao diện	44
2.8	Sơ đồ tuần tự cho chức năng tìm kiếm thông tin một chất dinh dưỡng khi người dùng cung cấp tên một chất dinh dưỡng cho service	44

2.9	Sơ đồ tuần tự cho chức năng tạo mới một thực đơn trên giao diện	45
2.10	Sơ đồ tuần tự cho chức năng tạo mới một thực đơn ở service	45
2.11	Sơ đồ tuần tự cho chức năng thêm một loại thực phẩm trên giao diện	46
2.12	Sơ đồ tuần tự cho chức năng sửa một loại thực phẩm trên giao diện	46
2.13	Sơ đồ tuần tự cho chức năng xóa một loại thực phẩm trên giao diện	47
2.14	Các thực thể Boundary, Control, Entity trong Class Diagram.	47
2.15	Class Diagram tìm kiếm thông tin hàm lượng dinh dưỡng . .	48
2.16	Class Diagram Tạo mới thực đơn	48
2.17	Cơ sở dữ liệu về dinh dưỡng	49
2.18	Cơ sở dữ liệu về quản lý hệ thống	49
2.19	Chức năng tìm kiếm thông tin.	56
2.20	Kết quả tìm kiếm thực phẩm.	56
2.21	Chức năng tạo mới thực đơn.	57
2.22	Chi tiết thực đơn.	57
3.1	Giao diện trang bìa	59
3.2	Giao diện nutrix web service	60
3.3	Giao diện khai thác dữ liệu cho nutrient	60
3.4	Giao diện trang quản lý hệ thống	61
3.5	Giao diện trang xem thông tin các thực phẩm	62
3.6	Giao diện trang xem thông tin chất dinh dưỡng	62

3.7	Giao diện trang sửa thông tin thực phẩm	63
3.8	Giao diện trang quản lý người dùng	64
3.9	Giao diện trang xem chi tiết thực phẩm	65
3.10	Giao diện trang tạo mới thực đơn	66
3.11	Giao diện trang chỉnh sửa thực đơn	66
3.12	Giao diện trang thêm mới thực phẩm trong thực đơn	67
3.13	Các giao diện xem thông tin thực phẩm, chất dinh dưỡng và xem chi tiết thông tin thực phẩm.	68
3.14	Giao diện hoàn thành cài đặt XAMPP	70

MỞ ĐẦU

1 Tính cấp thiết của đề tài

Trong thế kỷ 21, cải thiện tình trạng dinh dưỡng là trách nhiệm chung của mọi cá nhân, tổ chức, cộng đồng trong nước cũng như toàn thế giới nhằm chăm sóc sức khỏe cho bản thân, gia đình, mọi công dân, chống lại các nguy cơ bệnh tật liên quan đến chế độ dinh dưỡng không hợp lý. Chế độ dinh dưỡng không hợp lý hay thiếu hụt chất dinh dưỡng là một trong những nguyên nhân cơ bản dẫn đến các bệnh lý như suy dinh dưỡng, thừa cân, béo phì, cao huyết áp, bệnh lý về tim mạch, tiểu đường, các rối loạn khác dẫn đến ung thư và có thể gây nguy cơ tử vong. (xem[30])

Theo tổ chức Lương thực và Nông nghiệp Liên Hợp Quốc (Food and Agriculture Organization of the United Nations - FAO) trong năm 2010-2012 ước tính có gần 870 triệu người trên tổng số 7,1 tỷ người bị suy dinh dưỡng, tức là cứ tám người có một người bị suy dinh dưỡng (xem [8]). Các kết quả thống kê cho thấy, có 16 triệu người suy dinh dưỡng ở các nước đang phát triển (FAO 2012). Vấn đề dinh dưỡng không hợp lý góp đến một nửa lý do khiến 10,9 triệu trẻ em tử vong/1 năm. Sau năm 2008, có 1,4 tỷ người trên 20 tuổi bị thừa cân, nhìn chung hơn 10% dân số trưởng thành bị béo phì. Theo các chuyên gia, chiều cao và thể chất phụ thuộc 20% di truyền, 80% còn lại do dinh dưỡng, môi trường sống và rèn luyện thể thao (xem [?]). Theo Tổ chức

Y tế Thế giới (World Health Organization-WHO), một trong ba người ở các nước đang phát triển bị ảnh hưởng bởi sự thiếu hụt vitamin và khoáng chất. Việc thiếu chất dinh dưỡng như vitamin A, sắt, kẽm, Iot có thể dẫn đến thiếu máu, suy giảm tinh thần, năng lực nhận thức dẫn đến nguy cơ tử vong. Hiện trạng ở Việt Nam, theo số liệu từ tổng cục dinh dưỡng và viện thống kê tính đến năm 2013 có khoảng 2 triệu trẻ em dưới 5 tuổi bị suy dinh dưỡng thể thấp còi, với 15,3% suy dinh dưỡng nhẹ cân, 25,9% suy dinh dưỡng thấp còi và 6,6% gầy còm . Theo kết quả nghiên cứu trong vòng 10 năm (2000-2010) thì tỉ lệ thừa cân ở trẻ em dưới 5 tuổi là 4,9% và béo phì là 1,6%, tăng gấp 9 lần (xem [5]). Đa số bữa ăn gia đình Việt chưa đủ vi chất dinh dưỡng và chế độ dinh dưỡng không hợp lý. Tỷ lệ suy dinh dưỡng và thiếu vi chất ở trẻ em tại Việt Nam thuộc hàng cao nhất trên thế giới. 90% trẻ bị suy dinh dưỡng thấp còi trên thế giới tập trung ở 36 nước trong đó có Việt Nam.

Vấn đề đảm bảo dinh dưỡng hợp lý ngày càng được quan tâm ở Việt Nam và trên thế giới. Đã có rất nhiều kết quả nghiên cứu trong và ngoài nước đã nêu bật tầm quan trọng của việc đảm bảo dinh dưỡng bữa ăn hằng ngày cho cá nhân, gia đình, cộng đồng. Một số biện pháp đã được áp dụng trong nước như tăng cường mạng lưới dịch vụ, tư vấn và phục hồi dinh dưỡng, tuyên truyền vận động giáo dục dinh dưỡng trong trường học, cơ sở y tế. Theo “Chiến lược quốc gia về dinh dưỡng giai đoạn năm 2011-2020 và tầm nhìn đến 2030” thì mục tiêu phải đảm bảo dinh dưỡng cân đối, hợp lý là yếu tố nhằm hướng tới phát triển toàn diện về tầm vóc, thể chất trí tuệ người Việt Nam, nâng cao chất lượng cuộc sống (xem [6]). Có thể thấy trên thế giới đặc biệt là ở các quốc gia phát triển, đã công bố hàng loạt những nghiên cứu về thành phần dinh dưỡng trong thực phẩm có ảnh hưởng đến sức khỏe, có khả năng đẩy lùi bệnh tật

như cao huyết áp, tim mạch, phòng ngừa ung thư... (xem [25]).

Gần đây công nghệ thông tin nổi lên như một phương tiện hữu hiệu để thu thập và phổ biến thông tin về dinh dưỡng. Đẩy mạnh tin học hóa và xây dựng cơ sở dữ liệu chuyên ngành dinh dưỡng và an toàn thực phẩm sẽ tạo cơ hội để người dân tiếp cận thông tin về dinh dưỡng cũng như áp dụng các chế độ dinh dưỡng hợp lý. Trên thế giới, Bộ Nông nghiệp Hoa Kỳ (United States Department of Agriculture -USDA) đã tạo ra được nguồn cơ sở dữ liệu lớn hỗ trợ dinh dưỡng cho hàng trăm triệu người dân Mỹ (xem [32]). Cơ sở dữ liệu của USDA cung cấp chi tiết về danh sách chất dinh dưỡng và hàm lượng dinh dưỡng đã được đo đạc và kiểm định trong hàng nghìn loại thực phẩm khác nhau. Với sự kết hợp của khoa học dinh dưỡng và công nghệ thông tin, con người đã tạo ra được các phần mềm để đo đạc các chỉ số như chỉ số khối cơ thể (Body Mass Index-BMI) nhằm xác định người bị béo phì hay suy dinh dưỡng dựa vào chiều cao, cân nặng; Phần mềm phân tích lượng dinh dưỡng trong thức ăn nhằm cung cấp thông tin về hàm lượng dinh dưỡng. Như vậy, khi con người ngày càng nhận thức được mối quan hệ giữa dinh dưỡng và sức khỏe thì công nghệ thông tin hỗ trợ trong việc tạo ra các ứng dụng nhằm trợ giúp các chuyên gia dinh dưỡng và người sử dụng trong việc phân tích, kiểm soát chế độ ăn uống, lên thực đơn dinh dưỡng, đánh giá mức độ, tình trạng thể chất sức khỏe của mỗi người.

Từ nhu cầu thực tế của người sử dụng muốn có công cụ hỗ trợ trong việc kiểm soát tình trạng dinh dưỡng của bản thân, bài toán đặt ra là xây dựng hệ thống "Tra cứu thông tin và tạo thực đơn dinh dưỡng" với cơ sở dữ liệu tham khảo từ cơ sở dữ liệu dinh dưỡng của Bộ Nông nghiệp Hoa Kỳ có thể giúp người sử dụng tra cứu thành phần dinh dưỡng của thực phẩm, tạo thực đơn

dinh dưỡng để kiểm soát được lượng calories hấp thụ của cơ thể.

2 Mục tiêu nghiên cứu

Xây dựng hệ thống "Tra cứu thông tin và tạo thực đơn dinh dưỡng" giúp người dùng tra cứu thành phần dinh dưỡng của thực phẩm, tạo thực đơn dinh dưỡng để kiểm soát được lượng calories hấp thụ của cơ thể.

Để xây dựng được hệ thống này mục tiêu đặt ra như sau:

- Phân tích, xây dựng cấu trúc cơ sở dữ liệu dinh dưỡng; chuyển đổi để sử dụng cơ sở dữ liệu của bộ Nông nghiệp Hoa Kỳ.
- Tìm hiểu phương pháp Agile-TDD (Test Driven Development) để xây dựng và kiểm thử cho hệ thống.
- Xây dựng các User Story mô tả chính xác yêu cầu của người dùng đối với hệ thống.
- Nghiên cứu cách xây dựng các API với RESTful Web Service bằng Java.
- Xây dựng hệ thống hỗ trợ giao diện người dùng trên web, smartphone, các giao diện này được kết nối vào một dịch vụ web chung.
- Nghiên cứu và xây dựng ứng dụng phía giao diện người dùng sử dụng AngularJS.
- Nghiên cứu và xây dựng ứng dụng hỗ trợ trên điện thoại di động chạy trên nền tảng Android.

3 Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu:

- Công nghệ RESTful Web Service.
- Kỹ thuật xây dựng ứng dụng phía client với AngularJS.
- Kỹ thuật xây dựng ứng dụng trên thiết bị di động với Ionic framework

trên nền tảng Android.

- Bảo mật thông tin hệ thống.
- Xây dựng bộ test để kiểm thử phần mềm.

Trên cơ sở khảo sát và phân tích hệ thống, đề tài sẽ nghiên cứu và xây dựng hệ thống trong phạm vi những vấn đề sau:

- Đề tài được thực hiện trong lĩnh vực khoa học dinh dưỡng (các thông tin về thành phần chất dinh dưỡng) của Bộ Nông Nghiệp Hoa Kỳ.
- Thời gian nghiên cứu: 6 tháng (từ 1/11/2014 đến 15/4/2014).
- Công nghệ RESTful Web Service xây dựng các API.
- Tìm hiểu về xây dựng dịch vụ Web (Web Service) bằng Java.
- Sử dụng AngularJS xử lý giao diện người dùng.
- Bộ kiểm thử: phương pháp TTD (Test Driven Development).
- Kiểm tra bảo mật thông tin cho ứng dụng: Acunetix Web Vulnerability

4 Phương pháp nghiên cứu

- Phân tích và thiết kế hệ thống hướng đối tượng.
- Xây dựng cơ sở dữ liệu về dinh dưỡng.
- Xây dựng dịch vụ Web (Web Service) bằng Java.
- Tìm hiểu AngularJS để xây dựng giao diện người dùng.
- Cài đặt hệ thống.
- Kiểm tra bảo mật thông tin cho hệ thống.
- Ngoài ra sử dụng phương pháp nghiên cứu tài liệu, giáo trình có liên quan đến công nghệ cũng như nghiệp vụ của bài toán để xây dựng hệ thống, sử dụng hệ thống quản lý dự án để theo dõi và kiểm tra chất lượng hệ thống.

5 Bố cục đề tài

Nội dung chính của luận văn được trình bày trong ba chương. Cụ thể:

Chương 1: Cơ sở lý thuyết

Trình bày cơ sở lý thuyết và các công nghệ nền tảng sử dụng để xây dựng đề tài. Giới thiệu những công cụ sử dụng trong quá trình phát triển đề tài.

Chương 2: Phân tích và thiết kế hệ thống

Trình bày nội dung phân tích yêu cầu và đặc tả hệ thống dựa trên phân tích và thiết kế hướng đối tượng, bao gồm các chức năng, biểu đồ use case, class diagram, sequence diagram... và xây dựng cơ sở dữ liệu cho hệ thống.

Chương 3: Kết quả đạt được

Trình bày các giao diện của hệ thống tra cứu và xây dựng thực đơn dinh dưỡng.

Luận văn được hoàn thành tại trường Đại học Sư phạm Đà Nẵng dưới sự hướng dẫn của Tiến sĩ Nguyễn Trần Quốc Vinh.

Mặc dù đã rất cố gắng, song luận văn không tránh khỏi thiếu sót. Rất mong nhận được sự góp ý của các thầy cô và các bạn.

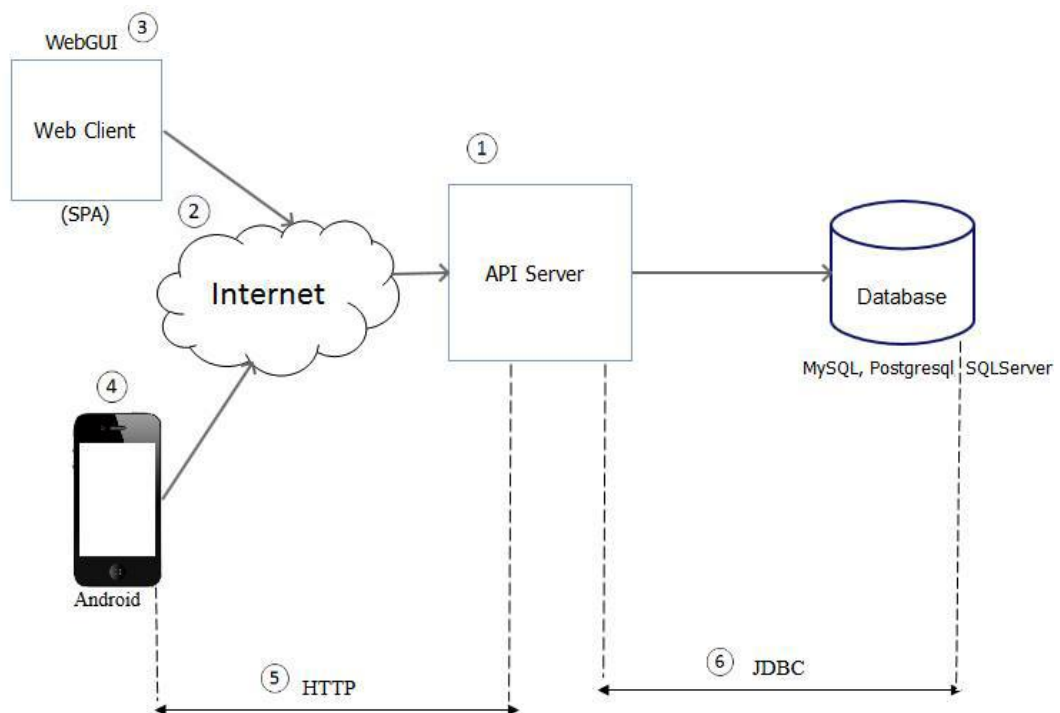
Xin chân thành cảm ơn!

Đà Nẵng, ngày 27 tháng 04 năm 2015

CHƯƠNG 1

CƠ SỞ LÝ THUYẾT

1 Sơ đồ tổng thể của hệ thống



Hình 1.1: Sơ đồ tổng quan của hệ thống

Trong đó:

1. API (Application Programming Interface) (xem [22]) là giao diện lập trình ứng dụng, cung cấp khuôn mẫu truy cập tài nguyên dưới hình thức dịch vụ web theo mô hình RESTful Web Service. Thông qua các API

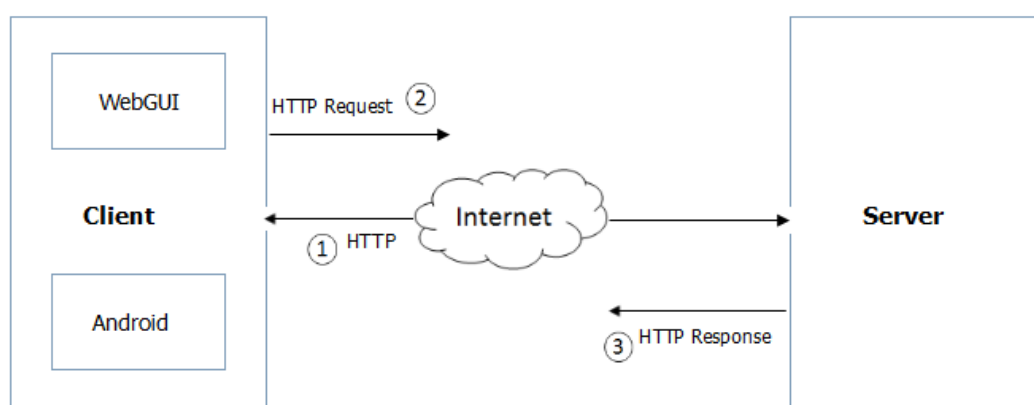
người lập trình có thể chỉnh sửa dữ liệu đang được lưu trữ trên cơ sở dữ liệu. Các chương trình phía Client (trang web dạng SPA, ứng dụng desktop, ứng dụng smartphone) đều kết nối và xử lý dữ liệu trên Server thông qua một API chung. Máy chủ cung cấp API, gọi là APIs Server, đơn giản là 1 ứng dụng dịch vụ web theo mô hình RESTful. APIs Server chờ Client gửi yêu cầu theo quy định trong API để gửi phản hồi cung cấp dữ liệu.

2. Mô hình RESTful Web Service (Representational State Transfer Web Service) (xem [38]) REST định nghĩa các quy tắc kiến trúc để thiết kế Web services chú trọng vào tài nguyên hệ thống, bao gồm các trạng thái tài nguyên được định dạng như thế nào và được chuyển tải qua HTTP. RESTful webservice có thể được viết bởi những ngôn ngữ lập trình khác nhau.
3. Web Client: sử dụng công nghệ SPA (Single Page Application) (xem [40]) “ứng dụng một trang” là ứng dụng web dành cho người dùng với giao diện chỉ có một trang duy nhất. Trong một SPA, chỉ bao gồm đúng một trang HTML và nhiều file JavaScript và CSS. Trang HTML tải về một lần duy nhất khi ứng dụng được nạp. Trang HTML sẽ không được tải lại bất kỳ thời điểm nào sau đó cũng như không chuyển đến một trang HTML khác, chỉ có dữ liệu được trao đổi với Server API thông qua AJAX. Nhờ vậy, tốc độ xử lý của ứng dụng web được nâng cao đáng kể.
4. Android:(xem [35]) là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện

thoại thông minh và máy tính bảng.

5. HTTP (Hypertext Transfer Protocol) (xem [37]) là giao thức truyền tải thông điệp trong mô hình Client/Server.
6. JDBC (Java Database Connectivity) (xem [19]) là một công nghệ chuẩn của nền tảng Java cho phép kết nối đến nhiều loại cơ sở dữ liệu quan hệ khác nhau. Nó cung cấp phương thức truy vấn và cập nhật dữ liệu theo ngôn ngữ chuẩn SQL. JDBC làm việc được với nhiều hệ điều hành như Windows, Mac OS, và các phiên bản khác nhau của Linux.

2 Giao thức làm việc của hệ thống



Hình 1.2: Giao thức làm việc của hệ thống

Giao tiếp giữa các thành phần ứng dụng trong hệ thống được thực hiện dựa trên công nghệ Restful Web Service với nền tảng là giao thức HTTP. Trong đó:

1. HTTP(xem [37]): là giao thức truyền thông độc lập dựa trên giao thức TCP/IP để truyền tải thông điệp dữ liệu giữa client/server. Giao thức HTTP quy định yêu cầu (gọi là HTTP Request) mà Client gửi đến cho

Server và cách thức Server gửi trả đáp ứng (gọi là HTTP Response) lại cho Client.

Client: là máy tính yêu cầu được cung cấp tài nguyên.

Server: là máy tính chuyên dụng dùng để cung cấp tài nguyên bằng cách gửi trả đáp ứng lại các yêu cầu của Client.

2. HTTP Request(xem [28]): Client gửi cho server một yêu cầu thực hiện công việc gì đó, server lần lượt gửi lại cho client kết quả mà client yêu cầu. Để thực hiện một yêu cầu thì HTTP Request bao gồm ba phần Method, Header, Body và có thêm URL.

2.1. METHOD

- GET: Phương thức GET được sử dụng để yêu cầu máy chủ lấy tài nguyên, theo định danh (id) được cung cấp.
- POST: Phương thức POST được sử dụng để yêu cầu máy chủ tạo ra một tài nguyên mới.
- PUT: Phương thức PUT được sử dụng để yêu cầu máy chủ chỉnh sửa/ cập nhật một tài nguyên hiện có, theo định danh (id) được cung cấp.
- DELETE: Phương thức DELETE được sử dụng để yêu cầu máy chủ xóa 1 nguồn tài nguyên, theo định danh (id) được cung cấp.
- HEAD: Tương tự như GET nhưng chỉ yêu cầu lấy phần tiêu đề.
- OPTIONS: Phương thức OPTIONS trả về phương thức HTTP mà server hỗ trợ cho các URL.
- PATCH: Phương thức PATCH được sử dụng để yêu cầu cập nhật

lại một phần tài nguyên.

2.2. Header: Cung cấp thêm thông tin cho yêu cầu gửi từ Client đến Server. Một số header có sẵn như: "Accept Charset", "Accept Language", "Accept Encoding", "Expect", "Host"...

2.3. URL (Uniform Resource Location): là địa chỉ duy nhất của một trang web, các tài nguyên được xác định bởi URL, gồm có Absolute URL (địa chỉ tuyệt đối) và Relative URL (địa chỉ tương đối).

2.4. Body: chứa dữ liệu mà Client gửi đến cho Server.

3. HTTP Response (xem [29]): Sau khi Server xử lý xong yêu cầu của Client, nó tạo ra thông điệp đáp ứng và gửi lại cho client. HTTP Response bao gồm Status Code (Mã trạng thái), Header, Body.

3.1. Status Code.

Có nhiều loại trạng thái trong giao thức HTTP:

- i. 1xx: Informational : yêu cầu được nhận và tiếp tục quá trình truy cập.
 - 100 Continue
 - Switching Protocols
- ii. 2xx: Success : hành động được nhận thành công.
 - 200 OK
 - 201 Created
- iii. 3xx: Redirection: Client phải có thêm hành động để hoàn thành yêu cầu.
 - 300 Multiple Choices

- 301 Moved Permanently
- iv. 4xx: Client Error: các yêu cầu không thể thực hiện được.
 - 400 Bad Request
 - 404 Not Found
- v. 5xx: Server Error: Máy chủ không thể thực hiện 1 yêu cầu hợp lệ.
 - 500 Internal Server Error
 - 501 Not Implemented

3.2. Header: Cung cấp thêm thông tin về cách thức xử lý tài nguyên trong HTTP Response khi gửi trả lại Client.

- Accept
- Accept-Charset
- Accept-Encoding
- Accept-Language
- Allow
- Connection
- Content-Base

3.3. Body: chứa các byte dữ liệu mà Server muốn gửi cho Client, nằm ngay sau các header.

3 RESTful Web Service

REST (Representational State Transfer Web Service) (xem [38],[17]) cho phép người dùng truy cập tài nguyên của ứng dụng web service thông qua URI. REST tuân thủ theo bốn nguyên tắc thiết kế cơ bản sau:

- Sử dụng các phương thức HTTP một cách rõ ràng.

- Phi trạng thái.
- Hiện thị cấu trúc thư mục như URLs.
- Định dạng theo JavaScript Object Notation (JSON) và XML hoặc cả hai.

REST yêu cầu các nhà phát triển ứng dụng phải sử dụng phương thức HTTP một cách rõ ràng theo cách tương thích với giao thức chuẩn. Nguyên lý thiết kế REST thiết lập một ánh xạ 1-1 giữa các hành động tạo, đọc, cập nhật và xóa (CRUD) với các phương thức HTTP (POST, GET,PUT, DELETE).

Cơ chế hoạt động:

- Client gửi yêu cầu cung cấp tài nguyên thông qua HTTP.
- Web service dựa trên yêu cầu để phản hồi lại kết quả trả về.
- Các phản hồi được trả về theo đúng định dạng yêu cầu của client.
- Client xử lý kết quả để đáp ứng yêu cầu của mình.

RESTful web service yêu cầu:

- HTTP method (GET, POST, PUT, DELETE)
- HTTP URI cho thấy tài nguyên.
- HTTP Header.
- HTTP entity-body.

RESTful web service trả về:

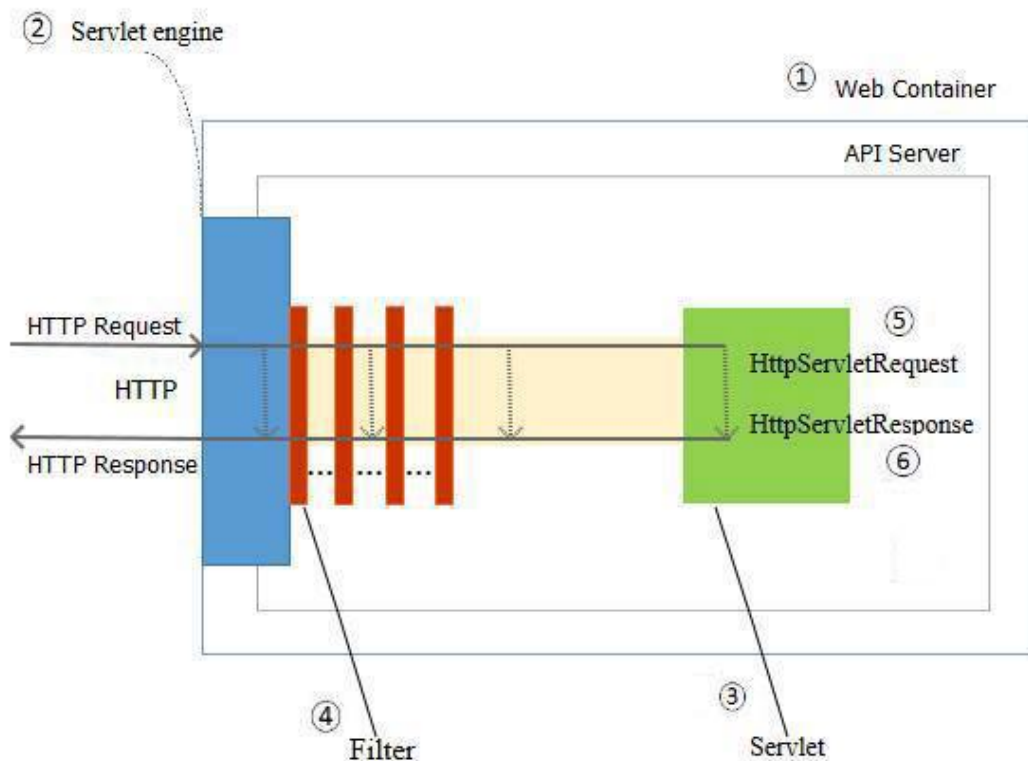
- XML
- JSON
- Text

4 Công nghệ JAX-RS

- JAX-RS (Java API for RESTful Web Service) (xem [16]) là một thư viện của Java nhằm mục đích hỗ trợ dịch vụ web theo RESTful.
- Sử dụng kiến trúc REST.
- JAX-RS sử dụng các chú thích (Annotation bases).(xem [21])
 - @Path: chỉ ra đường dẫn cho các lớp resource.
Ví dụ: @Path("/users/{username}")
 - @Produces: liệt kê các media type của dữ liệu được sinh ra.
 - @Consumes: liệt kê các media type mà dữ liệu đầu vào phải thỏa mãn.
 - @GET, @PUT, @POST, @DELETE và @HEAD: xác định các HTTP Request của các resource.
 - Ngoài ra cung cấp thêm các chú thích để lấy thông tin của các yêu cầu:
 - * @PathParam: liên kết với tham số phương thức để phân đoạn các đường dẫn.
 - * @QueryParam: liên kết với tham số phương thức để lấy các giá trị của tham số truy vấn HTTP.
 - * @HeaderParam: liên kết với tham số phương thức để lấy các giá trị của tham số tiêu đề HTTP.
 - * @MatrixParam: liên kết với tham số phương thức để lấy các giá trị của tham số ma trận HTTP.

- Implementation JAX-RS(xem [16]): Các thư viện của nhà cung cấp thứ ba gồm một số bản cài đặt phổ biến sau:
 - Oracle’s Jersey(xem [11]): cung cấp một API mở rộng từ JAX-RS với các tính năng bổ sung thêm nhằm đơn giản hóa dịch vụ RESTful sử dụng Java.
 - RESTlet(xem [39]): là một Framework phát triển cho các ứng dụng Java ,mã nguồn mở và hỗ trợ các định dạng dữ liệu và các tiêu chuẩn như HTTP, HTTPS, XML, JSON.
 - Apache CXF (xem [34]): là một thư viện mã nguồn mở, đầy đủ tính năng của Web Service. Nó kết hợp từ Celtix(xem [36])và XFire(xem [10]) thường được sử dụng với Apache ServiceMix, Apache Camel.
 - Jboss’s RESTEasy (xem [24]): là một dự án của JBoss cung cấp framework giúp xây dựng web Service và ứng dụng RESTful. Nó có thể chạy trên JBoss Application Server hoặc Servlet Container khác.

5 Servlet



Hình 1.3: Hoạt động của Servlet

Trong đó:

1. Web Container (xem [41]): là môi trường chạy cho các Web Applications. Nó xử lý các Servlet, tập tin JSP và các thành phần khác của Server. Web Container chịu trách nhiệm quản lý vòng đời của Servlet, thiết lập đường đi của một URL đến một Servlet.
2. Servlet Engine (xem [20]): Có chức năng đặc thù để tạo nội dung động. Khi Client gửi yêu cầu nội dung mà đó là nội dung động thì Servlet Engine có nhiệm vụ cung cấp lớp đối tượng Java (đối tượng Servlet) để xử lý và trả lại nội dung cho Client.

3. Servlet (xem [18]): là lớp các đối tượng Java được sử dụng để mở rộng khả năng truy cập của Server thông qua mô hình request-response. Mỗi một Servlet được ánh xạ với một URL, được người lập trình khai báo và được Container quản lý. Servlet có thể đáp ứng nhiều dạng ứng dụng khác nhau nhưng Servlet thường được sử dụng để xây dựng các ứng dụng trên web Server bằng Java. Công nghệ Java Servlet định nghĩa các lớp HTTP servlet cụ thể.

Packages `javax.servlet` và `javax.http` cung cấp các lớp để viết servlet.

Tất cả các servlets đều thừa kế từ Servlet interface, Class `HttpServlet` cung cấp phương thức như `doGet`, `doPost` để xử lý các dịch vụ HTTP cụ thể.

Vòng đời của Servlet:

- Web container khởi tạo một web component trước khi có một yêu cầu mới và gọi phương thức `init`. Phương thức `init` chỉ được gọi đúng một lần.
- Mỗi yêu cầu từ kết quả trình duyệt sẽ gọi đến một phương thức `service`.
- Container thông qua hai đối số của phương thức `service` để nhận yêu cầu và trả kết quả xử lý: một đối số đại diện cho HTTP request và một đối số đại diện cho HTTP response.
- Web container có thể xóa một thể hiện (instance) của web component. Nếu như web container xóa một thể hiện của web component thì gọi phương thức `destroy`.

4. Filter(xem [15]): là một đối tượng thực hiện nhiệm vụ lọc các yêu cầu đến resource hoặc trả lời phản hồi từ resource.

Phương thức filter: doFilter(), init(), destroy().

Filter API xác định bởi Filter, FilterChain, FilterConfig trong package javax.servlet.

5. HttpServletRequest: chứa dữ liệu truyền từ HTTP client đến HTTP Servlet. Nó được tạo bởi servlet container và được truyền cho servlet như một tham số của phương thức doGet() hoặc doPost(). HttpServletRequest mở rộng từ interface ServletRequest và cung cấp thêm các phương thức cho phép truy cập HTTP request URL, các thông tin về HTTP Request header, thông tin về Authentication type và User security, Cookies, Session.

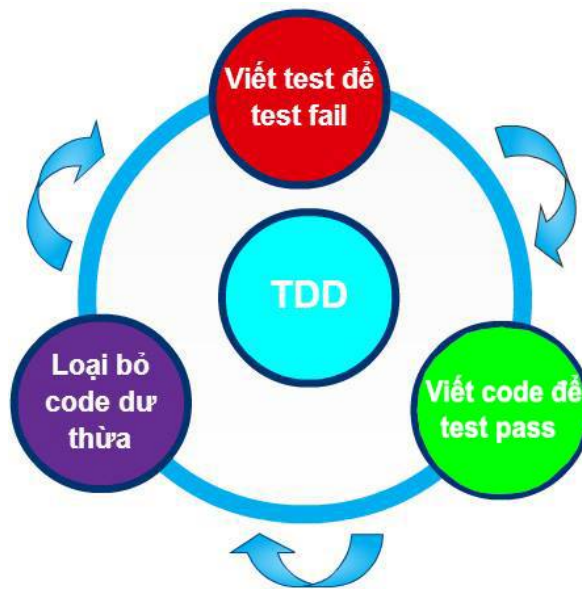
6. HttpServletResponse: chứa dữ liệu truyền từ HTTP servlet về HTTP client. Tất cả các servlet responses mở rộng giao diện lập trình ServletResponse. Nhiệm vụ của HTTP Servlet Response bao gồm:

- Lấy một output stream.
- Chỉ định content type.
- Có thiết lập buffer đầu ra không.
- Thiết lập localization information.
- Mã trạng thái HTTP trả về.
- Cookies.

Servlet là công nghệ chủ đạo của Java tạo ra nền tảng để xây dựng các công nghệ phát triển ứng dụng web khác bằng ngôn ngữ Java.

6 Phương pháp phát triển ứng dụng dựa trên mô hình TDD (Test-Driven Development)

- TDD (Test-Driven Development) là phương pháp tiếp cận mới để phát triển phần mềm trong đó có sự kết hợp giữa phương pháp Kiểm thử trước (Test First Development) (xem [33]) và phương pháp điều chỉnh lại mã nguồn (Refactoring) (xem [4]).
- TDD là một quy trình kỹ thuật phát triển phần mềm, thay vì viết mã rồi mới kiểm tra, với TDD người lập trình viên sẽ viết test kiểm tra chức năng trước khi viết code cho chức năng đó.
- Quy trình thực hiện kiểm thử theo TDD chú trọng đến các bước sau đây:
 - Bước 1. Add a Test: Viết một hàm test thử nghiệm nhỏ tương ứng với một chức năng, đảm bảo test đó lỗi. (vì chức năng chưa được lập trình nên ở bước này, test bị lỗi).
 - Bước 2. Write code: Viết code để vượt qua test.
 - Bước 3. Refactor: Cải thiện, Tối ưu hóa đoạn code của hàm vừa viết, đảm bảo test vẫn pass.
 - Bước 4. Lặp lại từ bước 1 cho các hàm khác.
- Chu trình của phương pháp TDD (xem hình 1.4)



Hình 1.4: Chu trình của Test Driven Development

- Ba quy tắc khi thực hiện TDD:(xem [31])
 - Không cho phép viết bất kỳ mã chương trình cho một chức năng nếu chưa có một unit test cho chức năng đó chuyển từ fail sang pass.
 - Không cho phép viết nhiều unit test cho nhiều chức năng cùng một lúc. (Trường hợp nhiều unit test cho một chức năng vẫn có thể chấp nhận được). Khi đã hoàn thành unit test, chuyển sang viết mã chức năng để pass unit test đó.
 - Không cho phép viết thêm mã chương trình của một chức năng khi nó đã đủ chuyển một unit test từ fail sang pass.
- Ưu điểm khi sử dụng phương pháp TDD (xem [3]).
 - Cải thiện chất lượng: TDD đề cao đến tính chính xác của phần mềm hơn bất kỳ các phương pháp tiếp cận nào trước đây. Nó cải thiện

chất lượng phần mềm bằng cách đòi hỏi việc viết các unit test trước và xem xét dưới góc độ của mã lệnh thực thi từ đó giảm bớt việc dò lỗi.

- Giảm thời gian: TDD rút ngắn thời gian test cho toàn hệ thống vì khi thực hiện các unit test sẽ hạn chế được mã lỗi sai của tổng thể hệ thống.
- Tăng năng suất tổng thể: Khi sử dụng TDD, việc gỡ lỗi là ít xảy ra vì người lập trình đã tập trung viết các unit test.
- Tăng độ tin cậy: Mỗi thay đổi nhỏ của hệ thống thì phải đảm bảo không ảnh hưởng đến các chức năng khác của hệ thống, TDD có thể làm được những việc sau: (1) cục bộ hóa các sai sót, (2) phát hiện các thay đổi ngoài ý muốn, (3) nâng cao tự tin cho người lập trình, (4) giảm thiểu nguy cơ thay đổi.
- Một thói quen áp dụng TDD vững chắc với nền tảng kiểm thử phong phú là sự chuẩn bị tốt nhất để đánh bại các đối thủ cạnh tranh, nắm bắt cơ hội: TDD tạo điều kiện cho vòng tích hợp phát triển ngắn hơn, hỗ trợ làm mịn nhanh chóng các yêu cầu mới, giúp cho người lập trình có thể cập nhật nhanh những thay đổi bất ngờ của sản phẩm.
- Sử dụng TDD không cần đến test hồi quy mới biết mã lệnh có đúng hay không: Nhờ TDD họ có thể biết ngay lập tức. Do đó TDD làm tăng sự tự tin và làm cho người lập trình có trách nhiệm hơn trong việc viết code.

- Nhược điểm của phương pháp TDD (xem [27]).
 - Cần có thời gian để thiết lập môi trường test và đào tạo cho người lập trình thích nghi với phương pháp TDD.
 - Phát triển các unit test đòi hỏi rất nhiều nỗ lực và kỹ năng.
 - Tự bản thân TDD không tăng cường chất lượng phần mềm, TDD chỉ giúp tập trung vào chỗ nào ta muốn cải thiện chất lượng. Muốn có chất lượng thì phải viết mã lệnh có chất lượng.
 - Gây ra hạn chế đối với một số lập trình viên có tư tưởng bảo thủ phong cách lập trình cá nhân: Đưa ra cấu trúc như TDD thì khác so với các nhà phát triển truyền thống. Bỏ ý thức lập trình cá nhân và chấp nhận thay đổi có thể giúp người lập trình nắm bắt các phương thức mới từ đó có thể tăng cường khả năng thích ứng của bản thân.
 - Với những dự án mới thì sẽ khởi động dự án chậm hơn do phải nỗ lực phát triển các unit test. Vì thành quả TDD đem lại thường đến chậm hơn trong chu trình phát triển phần mềm nên dễ khiến cho người phát triển ứng dụng cảm thấy mất niềm tin và cảm thấy việc viết test là lãng phí. Trong tình huống đó, nhóm trưởng cần giữ định hướng cho nhóm và luôn nhớ thành quả luôn đạt được cuối dự án.
 - Việc tiếp nhận TDD yêu cầu thay đổi ở cấp độ cá nhân của người lập trình. Trước khi áp dụng TDD, cần phải thay đổi cách nghĩ về phần mềm. TDD tác động đến cách thức bạn tiếp cận, viết và thiết kế phần mềm nên người lập trình cần thay đổi bản thân để thích nghi nhanh với TDD.

- Các công cụ phục vụ phương pháp TDD thường là các nền tảng cho kiểm thử mã nguồn mức đơn vị (unit test): JUnit, PHPUnit, NUnit, JMock, Mockito ...

Kiểm thử là một giai đoạn quan trọng trong quá trình phát triển phần mềm. Kiểm thử có thể tồn tại trước khi lập trình hoặc sau khi kết thúc lập trình. Với những ưu điểm của phương pháp TDD, luận văn áp dụng phương pháp này để xây dựng phần mềm.

7 Phương pháp phát triển ứng dụng web dựa trên mô hình SPA

7.1 Giới thiệu về SPA

Mô hình ứng dụng một trang SPA (Single Page Application)(xem [40]) là một dạng ứng dụng web dành cho người dùng cuối (end-users) với giao diện chỉ có một trang duy nhất. Trong một SPA, chỉ bao gồm đúng một trang HTML, nhiều file JavaScript và CSS. Trang HTML tải về lần đầu và duy nhất khi ứng dụng được nạp. Ứng dụng sẽ không tải lại trang vào bất kỳ thời điểm nào sau đó cũng như không chuyển đến một trang HTML khác. Hiện nay có rất nhiều khung ứng dụng hỗ trợ phát triển SPA, trong đó phải kể đến như là AngularJS, BackboneJS, EmberJS,...

7.2 Ưu và nhược điểm của SPA

7.2.1 Ưu điểm (xem [9])

- Xây dựng giao diện người dùng cuốn hút hơn: SPA cung cấp một trải nghiệm giao diện người dùng hấp dẫn hơn so với trang web trước đây.
- Giảm tải trên máy chủ: cách xây dựng trang web trước đây đòi hỏi phải nạp rất nhiều nội dung trùng lặp cho mỗi trang, với cách tiếp cận SPA tránh sự lặp lại và giúp ứng dụng hiệu quả hơn.

- Với SPA việc thay đổi hiển thị một vùng trên giao diện trở nên mượt mà hơn và không làm ảnh hưởng đến các vùng xung quanh.
- SPA giảm thiểu thời gian phản hồi của trang web bằng cách chỉ trao đổi dữ liệu với server. Trong thời gian chờ server phản hồi dữ liệu, giao diện sẽ hiển thị thanh tiến trình chờ giúp người dùng không cảm thấy ức chế vì sự chậm trễ.
- Cho phép người phát triển tự do hơn: khi tất cả nội dung đều có thể truy cập từ một trang, người lập trình sẽ dễ dàng trong việc bố trí sắp đặt nội dung trên trang đó.

7.2.2 Nhược điểm (xem[14])

- Hạn chế trong việc tối ưu hóa bộ máy tìm kiếm (Search Engine Optimization-SEO): SPA tạo giao diện và sắp đặt nội dung trên Client thông qua JavaScript nên bộ máy tìm kiếm không tiếp cận trang HTML đầy đủ khi thực hiện đánh chỉ mục trang. Tuy nhiên, hiện nay đã có giải pháp cung cấp trang web đầy đủ khi bộ máy tìm kiếm truy cập vào SPA (có thể tham khảo PhantomJS).
- Hiệu năng không đồng đều: SPA phụ thuộc rất nhiều vào phía Client, nghĩa là trang web của bạn có thể tải và hiển thị nhanh chóng với một số trình duyệt web (Client) trong khi một số khác chậm hơn do sử dụng thiết bị cũ.
- Caching: Vì các trang chỉ được sinh ra đầy đủ và sắp đặt nội dung vào giao diện hoàn chỉnh ở Client nên không thể cache đầy đủ các trang HTML ở phía Server, do đó chúng ta không thể tận dụng các kỹ thuật

Caching trong quá trình phát triển ứng dụng SPA.

8 AngularJS

Có nhiều khung ứng dụng (framework) khác nhau hỗ trợ xây dựng ứng dụng SPA. Nổi bật trong số đó mà chúng ta có thể kể đến là BackboneJS, EmberJS, AngularJS. Ứng dụng minh họa cho luận văn này sử dụng AngularJS để xây dựng phần giao diện SPA. AngularJS là dự án mã nguồn mở do Google đề xướng và chủ trì, bảo trợ phát triển. Hiện tại cho đến thời điểm này, AngularJS là khung ứng dụng SPA dẫn đầu về số lượng cộng đồng người dùng (xem [?])

AngularJS có cách tiếp cận khác biệt so với những cách thông thường. Bên cạnh việc sử dụng HTML làm ngôn ngữ tạo template, nó cho phép mở rộng HTML dưới dạng các directives giúp xử lý chức năng rõ ràng hơn.

8.1 Các khái niệm dùng trong AngularJS (xem [1])

- Directives: là các chỉ thị, dùng như thẻ hoặc thuộc tính mở rộng của HTML. Một số directives sử dụng trong AngularJS:
 - ng-app
 - ng-model
 - ng-bind
 - ng-init
 - ng-controller
- Model: Dữ liệu hiển thị cho người dùng xem.
- Scope: Phạm vi mà các model được lưu trữ trong controller, directive. Ngoài ra Scope đóng vai trò là ngữ cảnh của Expression, Scope

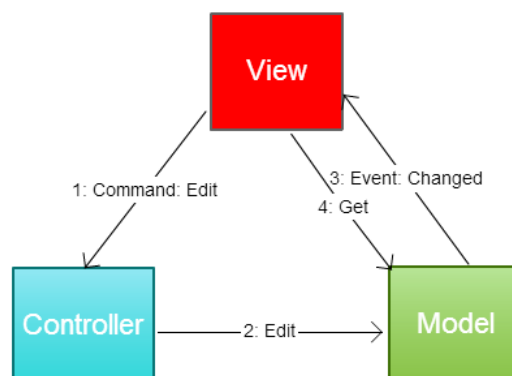
gắn kết dữ liệu 2 chiều với View, cung cấp cơ chế theo dõi thay đổi của trường dữ liệu...

- Expressions: Cấu trúc biểu thức dùng để truy cập biến và hàm được định nghĩa trong Scope.
- Compiler: Phân tích các khuôn mẫu, tạo đối tượng chỉ thị và biểu thức.
- Filter: Định dạng giá trị của một biểu thức để hiển thị cho người dùng.
- View: Là đối tượng DOM (Document Object Model), có thể chuyển đổi vùng dữ liệu để người dùng thấy.
- Data Binding: Đồng bộ dữ liệu giữa model và view.
- Controller: Đối tượng điều khiển, tiếp nhận và xử lý các tác động của người dùng từ giao diện. Mỗi Controller tương ứng với một scope chứa dữ liệu cho một vùng trên View.
- Module: Một container cho các thành phần khác nhau của một ứng dụng bao gồm các Controller, Service, Filter, Directive.
- Dependency Injection: Là mẫu hình thiết kế cô lập sự phụ thuộc giữa các đối tượng, đảm bảo các đối tượng không trực tiếp tạo ra các tham chiếu đến các đối tượng khác. Việc tạo các đối tượng và móc nối với nhau được ủy thác cho một Dependency Injection Container đảm nhiệm.
- Injector: Là Dependency Injection Container của AngularJS.
- Inject (bơm truyền): khái niệm để chỉ hành động lấy giá trị hoặc tham chiếu của một đối tượng (giả sử là A), gán cho thuộc tính của một đối tượng khác (đối tượng B) theo một cách nào đó.

- Provider (đối tượng cung cấp): là đối tượng do AngularJS tạo ra để chứa các định nghĩa về đối tượng dịch vụ. AngularJS cung cấp một số provider tạo sẵn cũng như cho phép nhà phát triển ứng dụng định nghĩa thêm các provider của riêng mình.
- Service (đối tượng dịch vụ): khái niệm này được AngularJS sử dụng để nói đến những đối tượng do AngularJS tạo ra, quản lý, và có thể được bơm truyền vào cho những thành phần khác.
- Instance of Service (thể hiện của đối tượng dịch vụ): khái niệm tương tự như đối tượng dịch vụ và được dùng lẫn lộn với khái niệm này trong AngularJS.

8.2 Mô hình MVC trong AngularJS

AngularJS hỗ trợ MVC trong thiết kế ứng dụng để chia rõ 3 thành phần chính là model (dữ liệu), view (giao diện), và controller (xử lý logic). Sử dụng mô hình MVC giúp các lập trình viên đơn giản hơn trong việc bảo trì, sửa lỗi, và nâng cấp hệ thống.



Hình 1.5: Mô hình MVC trong AngularJS

Trong đó (xem hình 1.5):

- Model: thành phần chứa dữ liệu, có thể nhận được từ file JSON.
- View: thành phần hiển thị giao diện trên màn hình.
- Controller: thành phần kiểm soát model và điều khiển view để hiển thị cho người dùng.

9 Xây dựng ứng dụng di động trên Ionic Framework

9.1 Giới thiệu về Ionic Framework

Ionic (xem [7]) là một framework dùng để phát triển ứng dụng hybrid trên điện thoại smart phone dựa trên HTML5. Một ứng dụng hybrid là một ứng dụng di động được xây dựng bằng các công nghệ phát triển web như HTML5, CSS, JavaScript và sau đó được đóng gói trong một môi trường giống như trình duyệt web (Native container) hiển thị nội dung có thể hoạt động trên hầu hết các thiết bị di động. Ứng dụng hybrid đem lại nhiều lợi thế như khả năng hiển thị nội dung trên tất cả các thiết bị di động, thời gian và chi phí dùng để tạo nên một ứng dụng hybrid cũng thấp hơn so với các ứng dụng di động thông thường.

Có thể coi Ionic như là một bộ khung front-end giúp người dùng kiểm soát hình ảnh và trải nghiệm trên ứng dụng di động.

Không như các framework khác như Famo.us, Framework7 hay OnsenUI, Ionic mang đến cho chúng ta những thành phần giao diện và cách bố trí mang phong cách rất giống với như phong cách thiết kế mặc định trên thiết bị di động.

Vì Ionic là một framework HTML5 nên nó cần bộ đóng gói (wrapper) như Cordova hay PhoneGap để có thể chạy được như một ứng dụng di động. Cordova là bộ đóng gói mặc định trong Ionic framework.

9.2 Cordova

Apache Cordova là một bộ khung để xây dựng ứng dụng di động sử dụng HTML, CSS và Javascript. Apache Cordova bao gồm một tập hợp các API thiết bị cho phép người lập trình di động truy cập, sử dụng các chức năng native của thiết bị như là camera hay cảm biến gia tốc bằng Javascript. Kết hợp với một bộ khung phát triển giao diện như jQuery Mobile or Dojo Mobile hoặc Ionic, cho phép ứng dụng di động có thể được phát triển chỉ dựa trên HTML, CSS và Javascript.

Cordova cung cấp một tập hợp các thư viện Javascript đã được chuẩn hóa để có thể sử dụng. Cordova hiện có thể sử dụng cho các nền tảng như iOS, Android, Blackberry, Windows Phone, Palm WebOS, Bada và Symbian.

Kể từ phiên bản 3.0, người lập trình có thể chọn một trong hai quy trình để phát triển một ứng dụng di động.

- Cross-platform workflow: được sử dụng khi muốn xây dựng ứng dụng trên càng nhiều nền tảng càng tốt, mà ít cần quan tâm đến đặc trưng của nền tảng.
- Platform-centered workflow: được sử dụng khi muốn tập trung phát triển ứng dụng trên một nền tảng cụ thể và muốn can thiệp sâu vào mức thấp ví dụ như khi muốn kết hợp một thành phần native đã được tùy biến vào một thành phần web của Cordova.

9.3 Tính năng nổi bật của Ionic Framework

9.3.1 Mã nguồn mở

Mục đích của Ionic framework không chỉ là xây dựng một bộ khung phát triển ứng dụng di động đa nền tảng mà còn muốn xây dựng một nền tảng để

chia sẻ các kiến thức cho các nhà phát triển, tạo ra một cách để đưa ra những design pattern tốt nhất để xây dựng các ứng dụng di động (xem [?]). Vì vậy, Ionic là một framework mở hoàn toàn cho phép tất cả mọi người có thể chia sẻ những kiến thức, kinh nghiệm để xây dựng những ứng dụng di động tuyệt vời trên nền tảng HTML5.

9.3.2 Cung cấp giao diện mẫu

Ionic framework không những cung cấp cho một số giao diện mẫu tốt mà còn cho khả năng kiểm soát tối đa trong quá trình phát triển ứng dụng.

9.3.3 Ứng dụng với scripting

Javascript trong Ionic có thể giúp người lập trình xây dựng những ứng dụng giống nhất với những ứng dụng native trên Android và iOS. Ionic hướng đến những tương tác giao diện mạnh mẽ như Side menu, Slide Box, Tab...

9.3.4 Sự kết hợp của Ionic và AngularJS

Ionic sử dụng AngularJS để tạo ra một framework cho việc xây dựng các ứng dụng di động đa nền tảng, nó cung cấp một bộ giao diện người dùng (User Interface) mã nguồn mở miễn phí đi cùng với các tính năng của AngularJS.

Việc xây dựng ứng dụng dựa trên AngularJS đòi hỏi mã nguồn phải có khả năng mở rộng cao để bổ sung các tính năng mới. Tuy nhiên với Ionic, người ta có thể tái sử dụng các chức năng trong ứng dụng trên các nền tảng khác nhau đồng thời vẫn có thể tùy chỉnh giao diện người dùng cho mỗi nền tảng riêng biệt.

9.3.5 Hiệu suất được cải thiện

Ionic rất chú trọng đến hiệu suất. Ionic có giao diện ổn định và tốc độ tốt, với các hiệu ứng chuyển động được áp dụng kỹ thuật tăng tốc phần cứng và tối giản các thao tác với DOM. Ionic cũng không cần sử dụng đến jQuery, mặc dù người sử dụng có thể thêm vào nếu muốn. Ionic có thể tạo ra các ứng dụng phức tạp chạy trên cả iOS và Android.

Kết luận

Chương 1 trình bày cơ sở lý thuyết của luận văn bao gồm các công nghệ sử dụng và phương pháp áp dụng để xây dựng hệ thống; sơ đồ tổng thể của hệ thống; công nghệ xây dựng ứng dụng giao diện Single-Paged Application bằng Angularjs kết nối với Server thông qua dịch vụ web Restful; phương pháp xây dựng hệ thống theo mô hình Agile - TDD (Test Driven Development), ngôn ngữ lập trình phía Client sử dụng AngularJS. Dựa trên cơ sở lý thuyết đã trình bày, chương 2 của luận văn sẽ tập trung vào việc phân tích và thiết kế hệ thống.

CHƯƠNG 2

PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

1 Khái quát nghiệp vụ bài toán

- **Dinh dưỡng** là quá trình cung cấp năng lượng từ thức ăn và chuyển hóa năng lượng trong tế bào để nuôi dưỡng cơ thể. Dinh dưỡng chiếm một vai trò quan trọng trong việc hình thành, phát triển cơ thể và giữ gìn sức khỏe của con người. Ở mỗi thời kỳ phát triển của đời người, nhu cầu về dinh dưỡng hoàn toàn khác nhau, tuy nhiên việc đáp ứng nhu cầu ấy một cách hợp lý lại luôn luôn là vấn đề đáng chú ý, vì đó là nền tảng của sức khỏe. Điều này càng đặc biệt quan trọng đối với trẻ nhỏ, vì những sai lầm về dinh dưỡng trong giai đoạn ấu thơ có khi gây những hậu quả nghiêm trọng và không thể phục hồi kéo dài đến suốt đời. Dinh dưỡng hợp lý còn có vai trò phòng ngừa bệnh và phục hồi sau bệnh (xem [42]).
- **Thức ăn** trong các bữa ăn hàng ngày được chế biến từ nhiều loại thực phẩm khác nhau. Hệ tiêu hóa sẽ phân giải các loại thực phẩm này thành các **chất dinh dưỡng**. Chỉ có chất dinh dưỡng mới được hấp thu vào máu.
- Có hơn 40 chất dinh dưỡng cần thiết cho cơ thể chia làm 3 nhóm chính:

1. Nhóm chất dinh dưỡng đa lượng sinh năng lượng

- Chất bột đường (carbohydrate, glucid);
- Chất béo (Lipid);
- Chất đạm (Protein): có 8 acid amin thiết yếu là những acide amin bắt buộc phải đưa vào cơ thể qua thực phẩm vì cơ thể không tự tổng hợp được (Histidin, Isoleucine, Leucine, Lysine, Methionine, Phenylalanine, Threonin, Tryptophan, Valine, Taurine).

2. Nhóm chất dinh dưỡng vi lượng

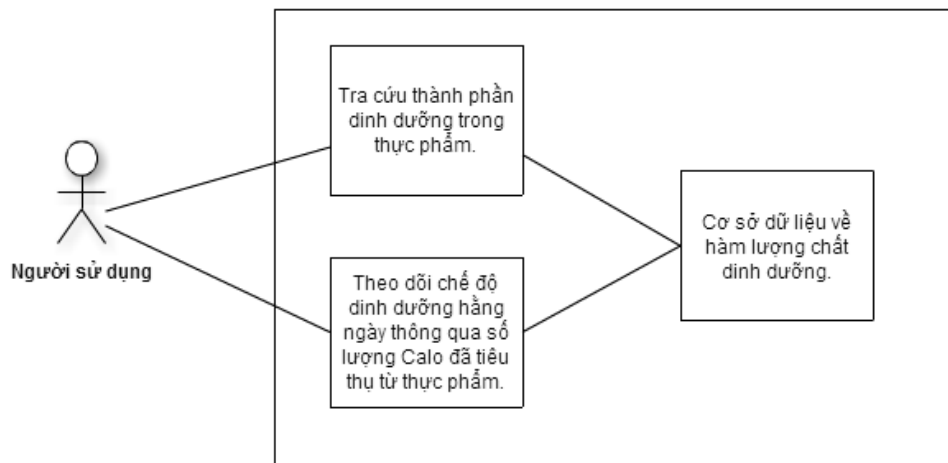
- Vitamin: Gồm các vitamin tan trong nước (B, C) và các vitamin tan trong chất béo (A, D, E, K);
- Chất khoáng vi lượng: Hiện đã xác định được khoảng 10 loại khoáng chất vi lượng hiện diện trong cơ thể nhưng chỉ mới biết được chức năng và chuyển hóa của Zn, Fe, Mg, Cu, I, F, Se.

3. Nhóm chất dinh dưỡng đa lượng không sinh năng lượng

- Chất khoáng đa lượng: Canxi, Phosphor, Potassium, Sulfur, Sodium, Chloride, Magnesium;
- Chất xơ: Không tiêu hóa, không hấp thu nhưng có vai trò quan trọng trong điều hòa hoạt động của hệ tiêu hóa;
- Nước: Là thành phần chính yếu của khẩu phần dù ít được quan tâm.

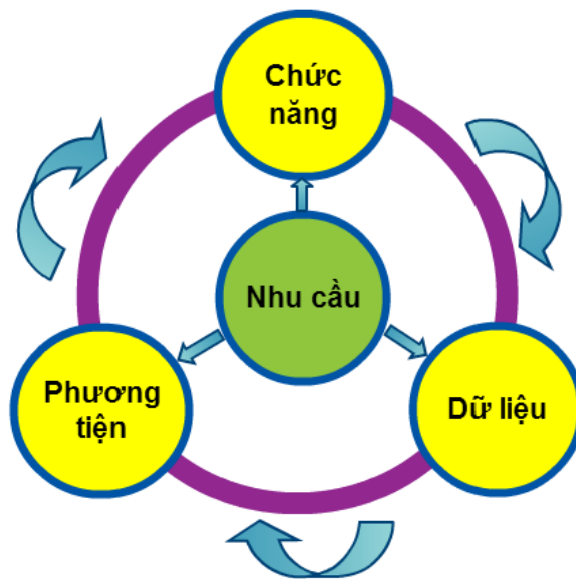
2 Hướng tiếp cận bài toán

Từ nhu cầu thực tế của người sử dụng muốn có công cụ hỗ trợ trong việc kiểm soát tình trạng dinh dưỡng của bản thân, bài toán đặt ra là xây dựng hệ thống "Tra cứu thông tin và tạo thực đơn dinh dưỡng" với cơ sở dữ liệu tham khảo từ cơ sở dữ liệu dinh dưỡng của Bộ Nông nghiệp Hoa Kỳ có thể giúp người sử dụng tra cứu thành phần dinh dưỡng của thực phẩm, tạo thực đơn dinh dưỡng để kiểm soát được lượng calories hấp thụ của cơ thể.



Hình 2.1: Bài toán đặt ra của hệ thống

Sơ đồ 2.2 mô tả các phương diện của hệ thống. Xuất phát từ nhu cầu bài toán thì các chức năng, dữ liệu và phương tiện là những phương diện tương tác lẫn nhau để xây dựng hệ thống hoàn chỉnh.



Hình 2.2: Sơ đồ mối quan hệ giữa các phương diện của hệ thống.

Về phương diện mô tả nhu cầu, nhu cầu của người sử dụng được mô tả ngắn gọn bằng các bảng tóm lược yêu cầu người dùng (user story)(xem [2]),theo cấu trúc “*As a* <type of user>, *I want*<some goal> *so that* <some reason>”. User story thường dùng với phương pháp phát triển phần mềm linh hoạt (Agile software development) (xem [26]). Các user story làm cơ sở cho việc xác định các chức năng của hệ thống một cách nhanh chóng.

Ví dụ về user story của hệ thống:

- **Là một** người dùng, **tôi muốn** ứng dụng có chức năng tìm kiếm thông tin hàm lượng chất dinh dưỡng có trong một loại thực phẩm khi nhập tên thực phẩm vào ô tìm kiếm **để** có thể truy vấn hàm lượng chất dinh dưỡng.
- **Là một** người dùng, **tôi muốn** ứng dụng có thể so sánh được thông tin về giá trị dinh dưỡng của hai loại thực phẩm khác nhau do tôi chọn từ

danh sách lọc tìm được **để** tôi đưa ra lựa chọn nên dùng loại thực phẩm nào.

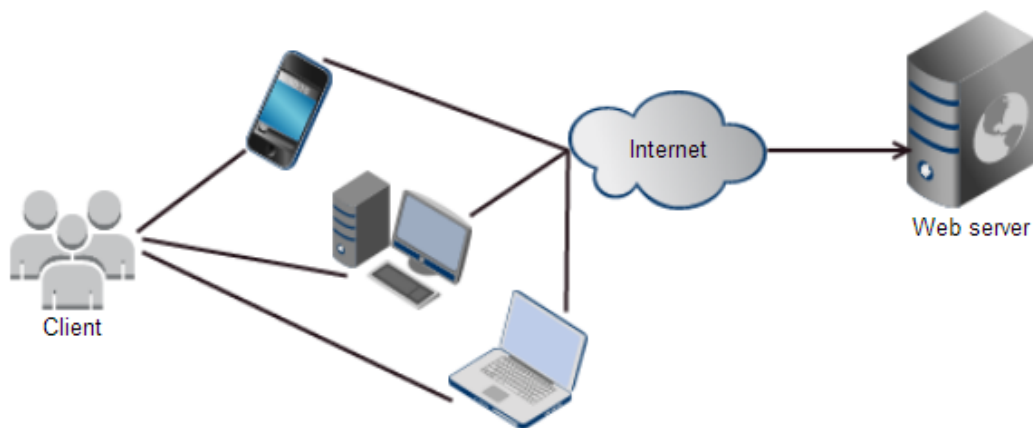
- **Là một** người dùng, **tôi muốn** ứng dụng có chức năng cho phép tạo thực đơn **để** có thể theo dõi lượng chất dinh dưỡng, calories mà bản thân đã sử dụng.
- **Là một** người quản trị hệ thống, **tôi muốn** ứng dụng cho phép tạo mới, cập nhật thông tin chất dinh dưỡng, thực phẩm **để** phù hợp với cơ sở dữ liệu của hệ thống.

Từ các mô tả nhu cầu người dùng đó ta có thể xác định được **chức năng** tổng quan của hệ thống được nêu ở phần 3

Ngoài ra với các bảng nhu cầu (user story) chúng ta dễ dàng viết các bộ kiểm thử chấp nhận (acceptance test) để kiểm tra toàn diện tính năng của ứng dụng.

Về phương diện dữ liệu, hệ thống sử dụng cơ sở dữ liệu được thu thập từ nguồn dữ liệu của Bộ Nông nghiệp Hoa Kỳ- USDA Food Composition Database, version 26, năm 2013 (cơ sở dữ liệu với hơn 8000 thực phẩm kèm theo thông tin dinh dưỡng) (xem[13]; sách *Thành phần thực phẩm Việt Nam* của Bộ Y tế và Viện Dinh dưỡng Việt Nam biên soạn, xuất bản năm 2007 (xem[23])). Cơ sở dữ liệu của hệ thống được tổ chức dưới dạng các bảng cơ sở dữ liệu quan hệ.

Về phương diện phương tiện, hệ thống hỗ trợ giao diện người dùng trên smartphone, giao diện web và có khả năng mở rộng giao diện trên desktop. Các giao diện này đều được kết nối vào một dịch vụ web chung.



Hình 2.3: Các phương tiện hệ thống sử dụng.

3 Chức năng của hệ thống

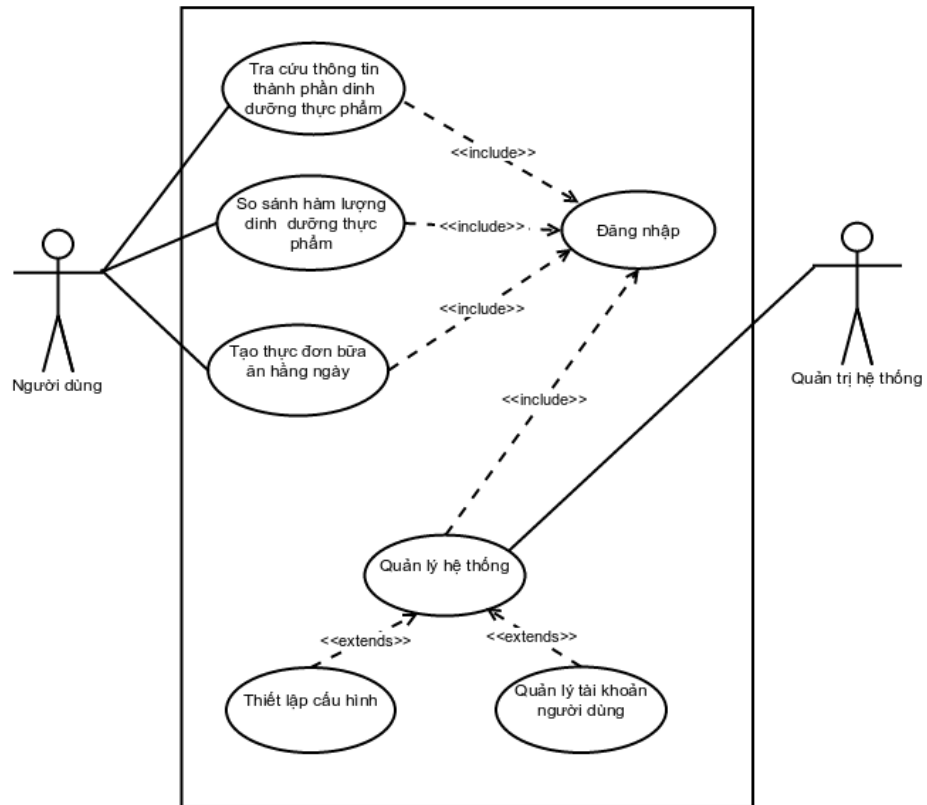
- Cung cấp cho người dùng thông tin về hàm lượng dinh dưỡng khi người dùng nhập một loại thực phẩm.
- So sánh hàm lượng dinh dưỡng trong thực phẩm.
- Tạo thực đơn bữa ăn hằng ngày theo nhu cầu năng lượng dinh dưỡng.

4 Các tác nhân của hệ thống

Tên tác nhân	Mô tả
Admin	Quản lý hệ thống, tạo tài khoản người dùng, cập nhật dữ liệu cho hệ thống.
Người dùng	Truy cập vào trang web và tra cứu thông tin và tạo thực đơn dinh dưỡng.

Bảng 2.1: Tác nhân của hệ thống

5 Lược đồ Usecase của toàn hệ thống



Hình 2.4: Lược đồ usecase của hệ thống.

5.1 Mô tả Usecase dưới hình thức User Story

- **Là một** người dùng, **tôi muốn** ứng dụng có chức năng tìm kiếm thông tin về hàm lượng chất dinh dưỡng có trong một loại thực phẩm, bằng cách hiển thị ô nhập tên loại thực phẩm cần tra cứu **để** khi nhập tên một loại thực phẩm có trong CSDL, ứng dụng sẽ hiển thị ra màn hình thông tin hàm lượng dinh dưỡng dưới dạng một bảng danh sách các chất dinh dưỡng kèm theo hàm lượng có trong thực phẩm đó.
- **Là một** người dùng, **tôi muốn** ứng dụng có chức năng so sánh được

thông tin về giá trị dinh dưỡng của hai loại thực phẩm khác nhau, bằng cách hiển thị ô nhập tên loại thực phẩm cần so sánh **để** khi nhập tên hai loại thực phẩm có trong CSDL, ứng dụng sẽ hiển thị ra màn hình một bảng so sánh thông tin hàm lượng dinh dưỡng với các chất dinh dưỡng kèm theo hàm lượng có trong thực phẩm và sự khác nhau giữa hai loại thực phẩm sẽ được thể hiện bởi chữ có màu in đậm nếu hàm lượng dinh dưỡng trong chất đó lớn hơn hàm lượng dinh dưỡng trong chất còn lại.

- **Là một** người dùng, **tôi muốn** ứng dụng có chức năng cho tôi tự chọn thực đơn bằng cách hiển thị các ô nhập tên thực phẩm; số lượng người sử dụng thực đơn mong muốn; **để** khi tôi chọn thực đơn các món ăn có trong CSDL, ứng dụng sẽ hiển thị ra màn hình một danh sách các món ăn tôi đã chọn kèm theo thành phần dinh dưỡng, định lượng cho từng thực phẩm với các chức năng thêm, sửa, xóa thực phẩm, đồng thời tính được năng lượng đã tiêu thụ của thực đơn đó nhằm giúp tôi kiểm soát được thực đơn dinh dưỡng.

5.2 Mô tả Acceptance Test

Acceptance Tests (Customer Tests hay Customer Acceptance Tests) (xem [?]) là một bước kiểm thử được thực hiện bởi khách hàng sử dụng phần mềm. Khách hàng sẽ dựa trên User Story của hệ thống hoặc tự xác định kịch bản để kiểm tra yêu cầu họ đưa ra có được người phát triển phần mềm thực hiện chính xác hay chưa và khi có một chức năng cần thay đổi thì đảm bảo được nó không ảnh hưởng đến các chức năng khác. Một User Story có thể có một hoặc nhiều Acceptance Test. Acceptance Test là một dạng kiểm thử hộp đen (black-box system test).

Mục tiêu của Acceptance Test là thể hiện sự chấp nhận của khách hàng với hệ thống.

Một Acceptance Test chạy được với dữ liệu đầu vào do khách hàng đưa ra, sau đó kết quả thu được so sánh với kết quả mong đợi. Nếu tất cả trường hợp đều đúng thì vượt qua được bộ kiểm tra.

Minh họa bên dưới mô tả một Acceptance Test cho một User Story:

Là một người dùng, **tôi muốn** ứng dụng có chức năng tìm kiếm thông tin về hàm lượng chất dinh dưỡng có trong một loại thực phẩm, bằng cách hiển thị ô nhập tên loại thực phẩm cần tra cứu **để** khi nhập tên một loại thực phẩm có trong CSDL, ứng dụng sẽ hiển thị ra màn hình thông tin hàm lượng dinh dưỡng dưới dạng một bảng danh sách các chất dinh dưỡng kèm theo hàm lượng có trong thực phẩm đó.

Bảng mô tả Acceptance Test:

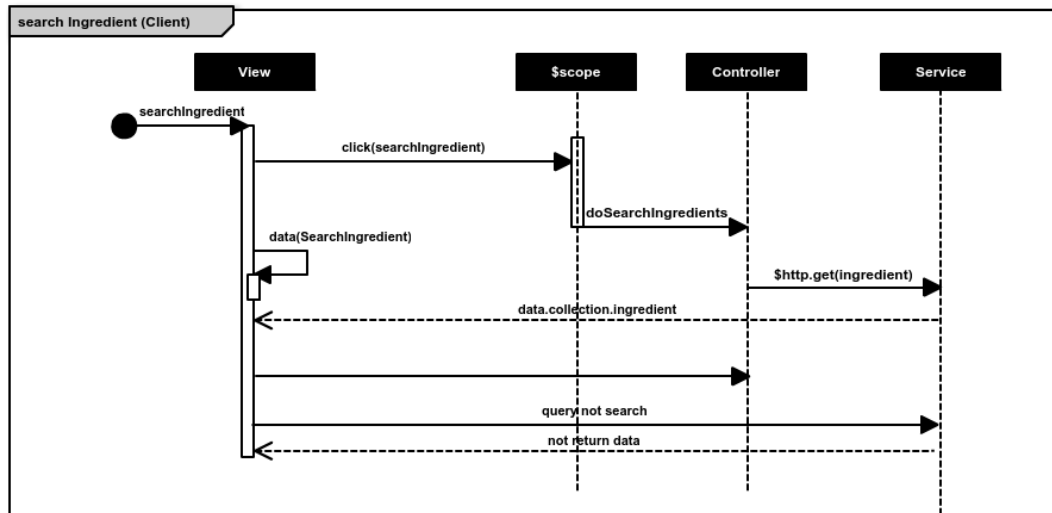
Mô tả	Màn hình tìm kiếm thành phần dinh dưỡng trong thực phẩm cho phép người dùng xem thông tin thành phần dinh dưỡng trong một loại thực phẩm.
Cài đặt	1. Nhập tên thực phẩm khớp với chuỗi tìm kiếm. 2. Nhập tên thực phẩm không tồn tại trong cơ sở dữ liệu. 3. Nhập tên thực phẩm chứa kí tự đặc biệt, số, khoảng trắng giữa các kí tự.
Hướng dẫn	1. Hiển thị màn hình tìm kiếm. 2. Nhập vào tên 1 loại thực phẩm bắt đầu bằng "%C*" 3. Nhấn nút tìm kiếm.

Kết quả mong đợi	<p>1. Hiển thị kết quả sau trong hộp kết quả tìm kiếm:</p> <p>Cà chua</p> <p>Cà tím</p> <p>Cá ngừ</p> <p>...</p> <p>2. Người dùng chọn một loại thực phẩm để xem thông tin thành phần dinh dưỡng có trong loại thực phẩm đã chọn.</p> <p>3. Tên thực phẩm không tồn tại trong cơ sở dữ liệu chẳng hạn nhập tên thực phẩm cần tìm là "joseph", màn hình phải hiện thông báo không tìm thấy thực phẩm có tên là "joseph".</p> <p>4. Tên thực phẩm chứa kí tự đặc biệt, số, khoảng trắng giữa các kí tự chẳng hạn như "p o r k", màn hình phải hiện thông báo yêu cầu nhập lại tên thực phẩm và thông báo không tồn tại tên thực phẩm trong cơ sở dữ liệu.</p>
------------------	---

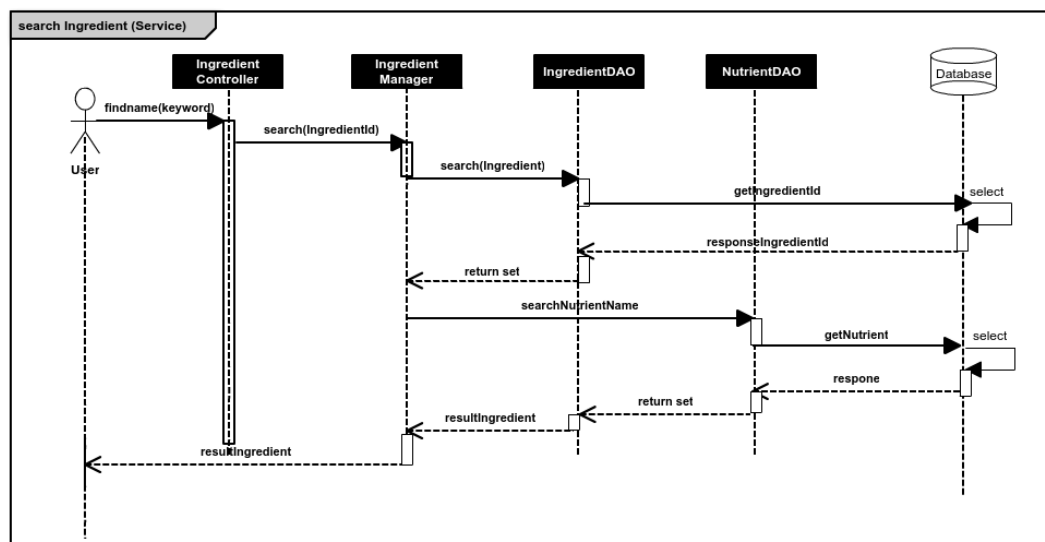
Bảng 2.2: Bảng mô tả Acceptance Test cho chức năng tìm kiếm thành phần dinh dưỡng trong thực phẩm

6 Sequence Diagram

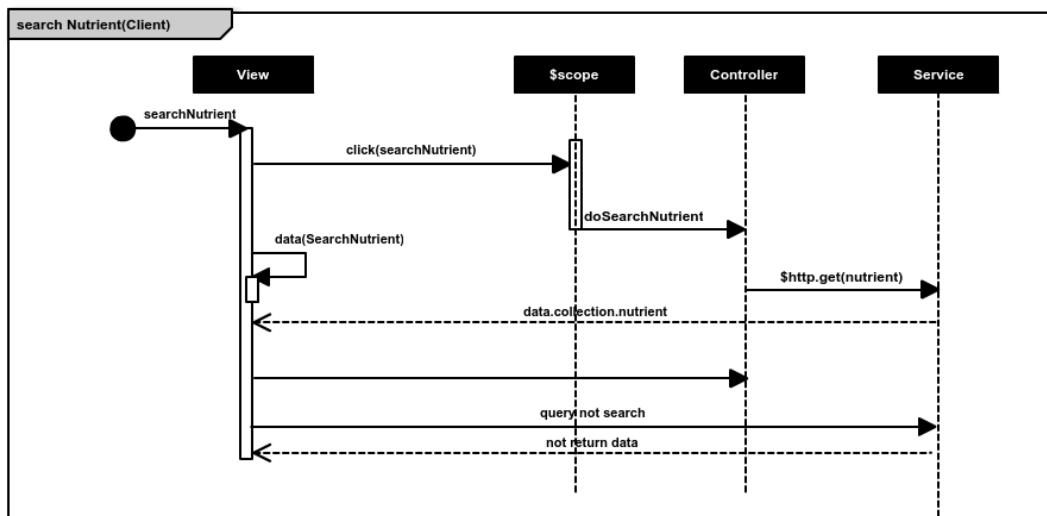
Các Sequence Diagram của hệ thống.



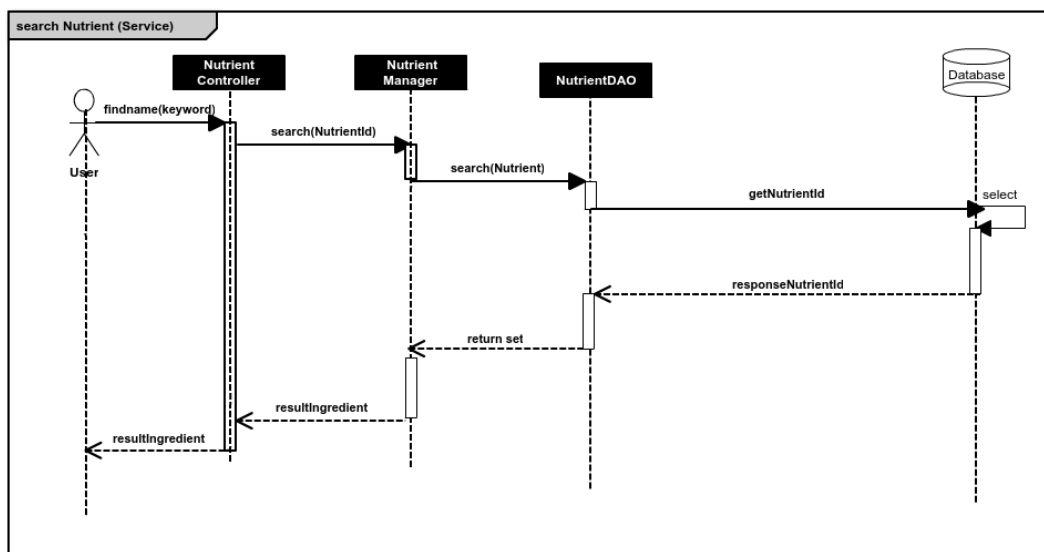
Hình 2.5: Sơ đồ tuần tự cho chức năng tìm kiếm thông tin của một loại thực phẩm trên giao diện



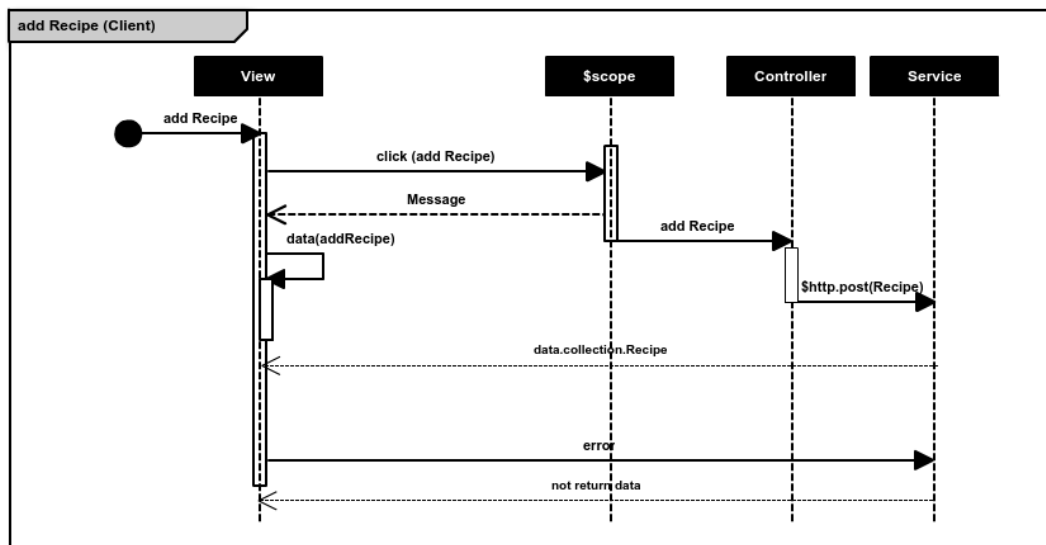
Hình 2.6: Sơ đồ tuần tự cho chức năng tìm kiếm thông tin loại thực phẩm khi người dùng cung cấp tên một loại thực phẩm cho service



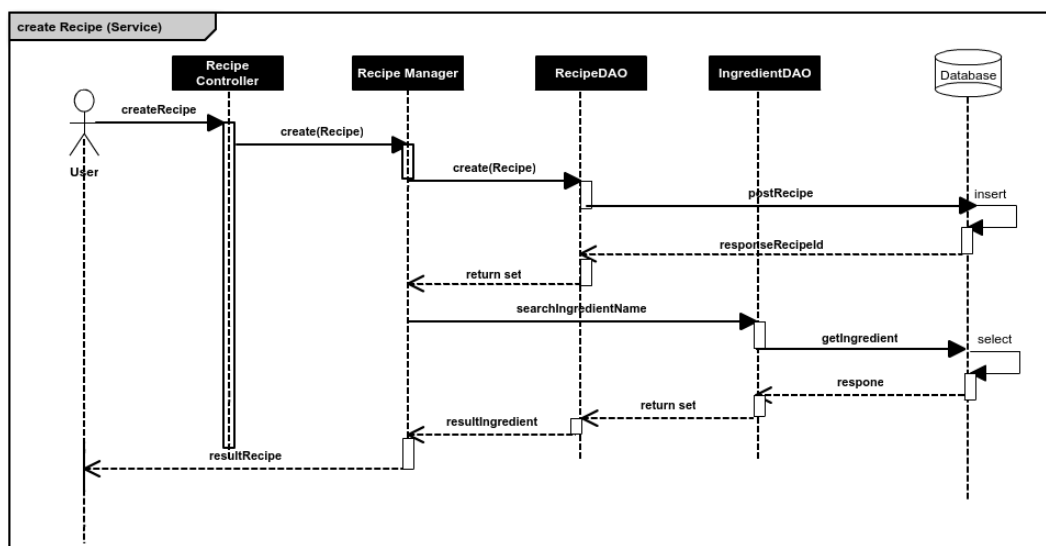
Hình 2.7: Sơ đồ tuần tự cho chức năng tìm kiếm thông tin của một chất dinh dưỡng trên giao diện



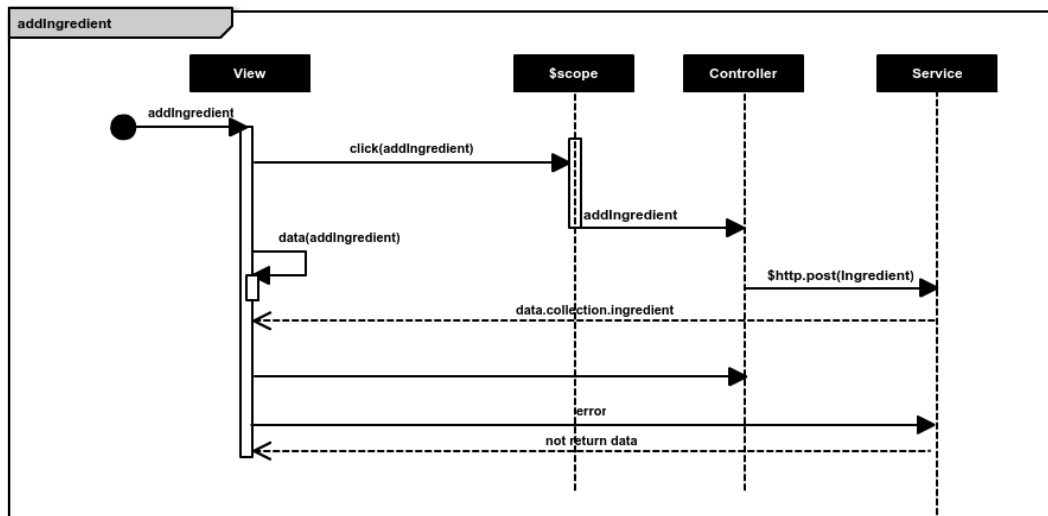
Hình 2.8: Sơ đồ tuần tự cho chức năng tìm kiếm thông tin một chất dinh dưỡng khi người dùng cung cấp tên một chất dinh dưỡng cho service



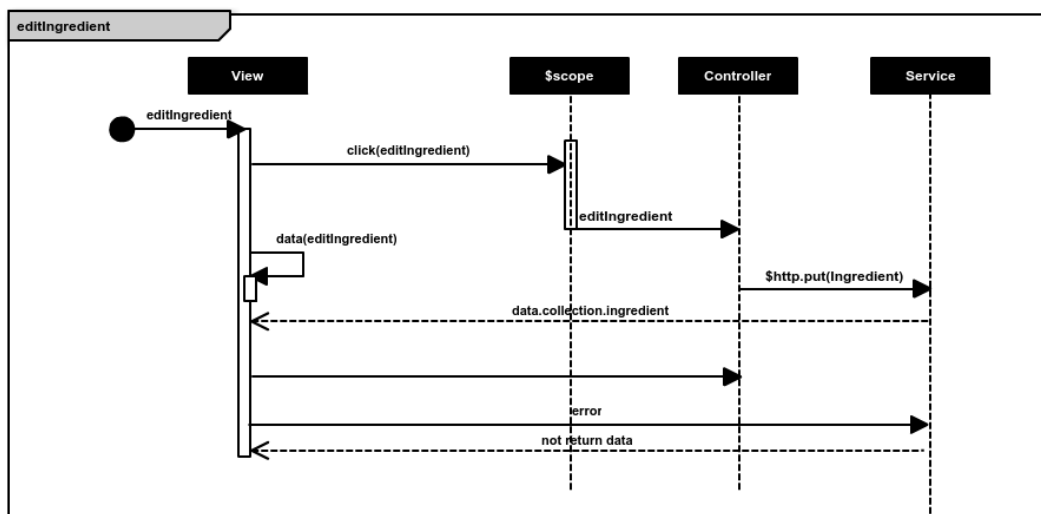
Hình 2.9: Sơ đồ tuần tự cho chức năng tạo mới một thực đơn trên giao diện



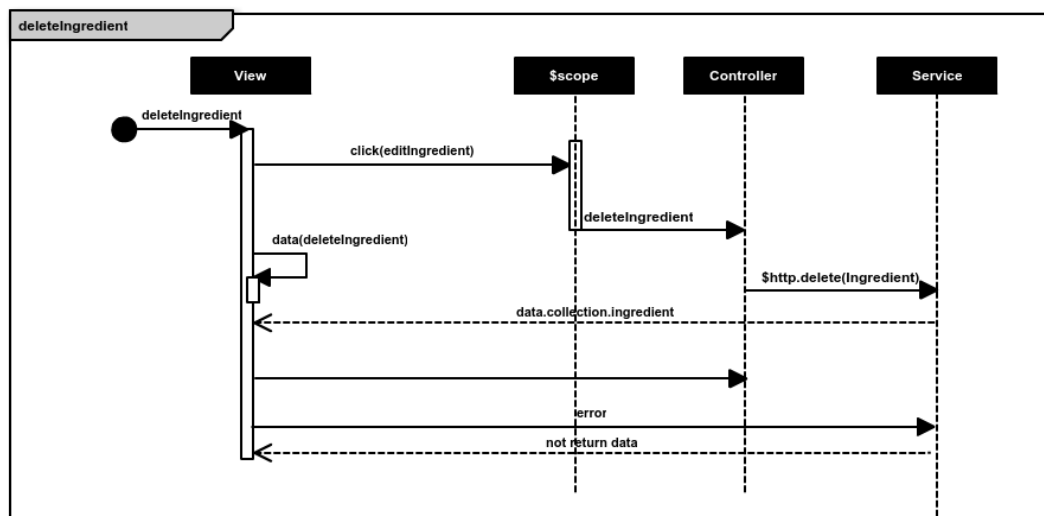
Hình 2.10: Sơ đồ tuần tự cho chức năng tạo mới một thực đơn ở service



Hình 2.11: Sơ đồ tuần tự cho chức năng thêm một loại thực phẩm trên giao diện



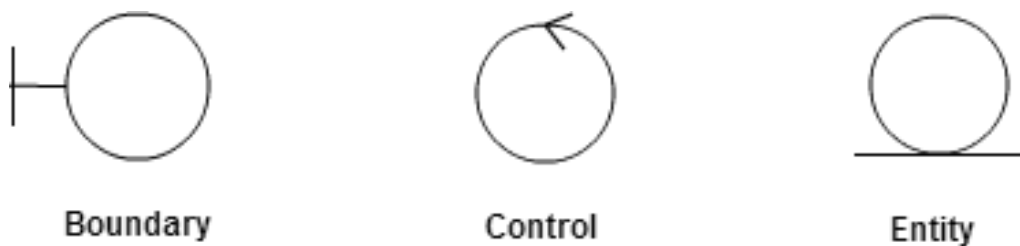
Hình 2.12: Sơ đồ tuần tự cho chức năng sửa một loại thực phẩm trên giao diện



Hình 2.13: Sơ đồ tuần tự cho chức năng xóa một loại thực phẩm trên giao diện

7 Class Diagram

Class Diagram chia làm 3 loại thực thể là Boundary, Control và Entity.

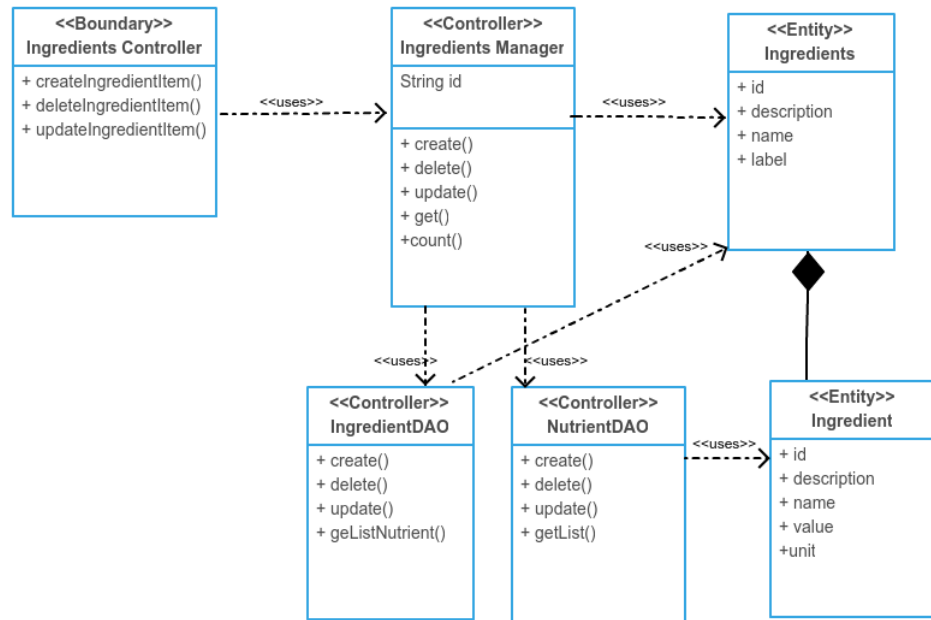


Hình 2.14: Các thực thể Boundary, Control, Entity trong Class Diagram.

Trong đó:

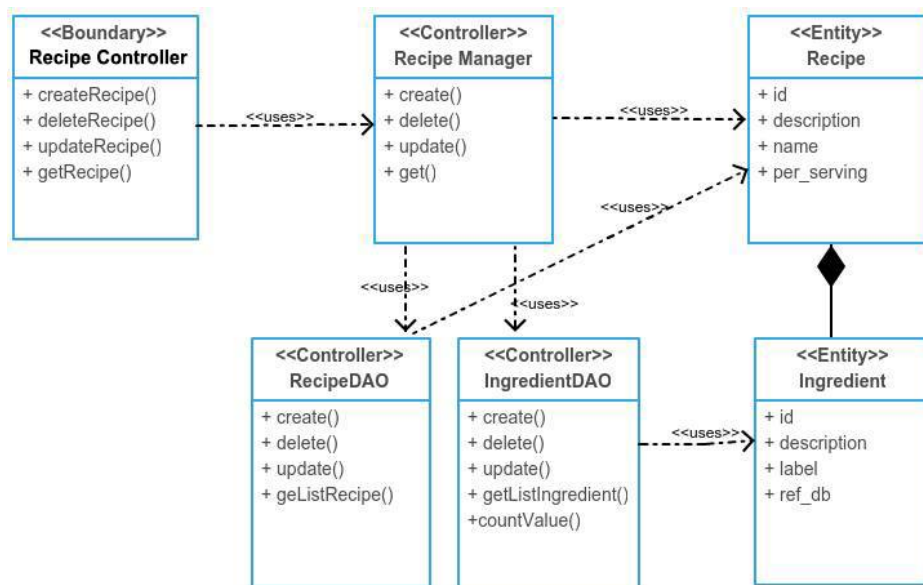
- Entity: đối tượng mô hình, có chức năng lưu trữ thông tin về nghiệp vụ.
- Boundary: nhóm các đối tượng giao tiếp với hệ thống và môi trường bên ngoài.
- Control: đối tượng trung gian giữa Entity và Boundary, thực thi các lệnh do Boundary gửi đến và gửi trả dữ liệu từ Entity về Boundary.

Class diagram cho chức năng tìm kiếm thông tin hàm lượng dinh dưỡng khi người dùng nhập tên thành phần thực phẩm.



Hình 2.15: Class Diagram tìm kiếm thông tin hàm lượng dinh dưỡng

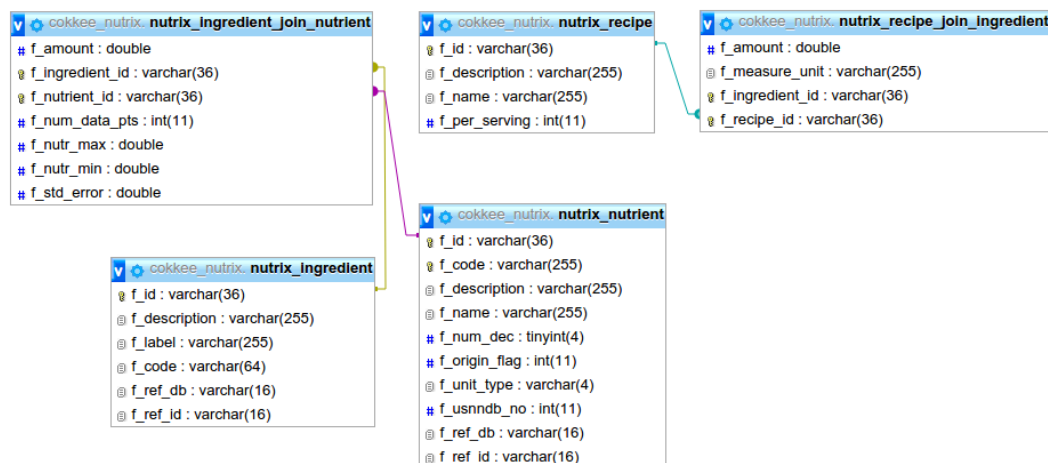
Class diagram cho chức năng tạo mới thực đơn.



Hình 2.16: Class Diagram Tạo mới thực đơn

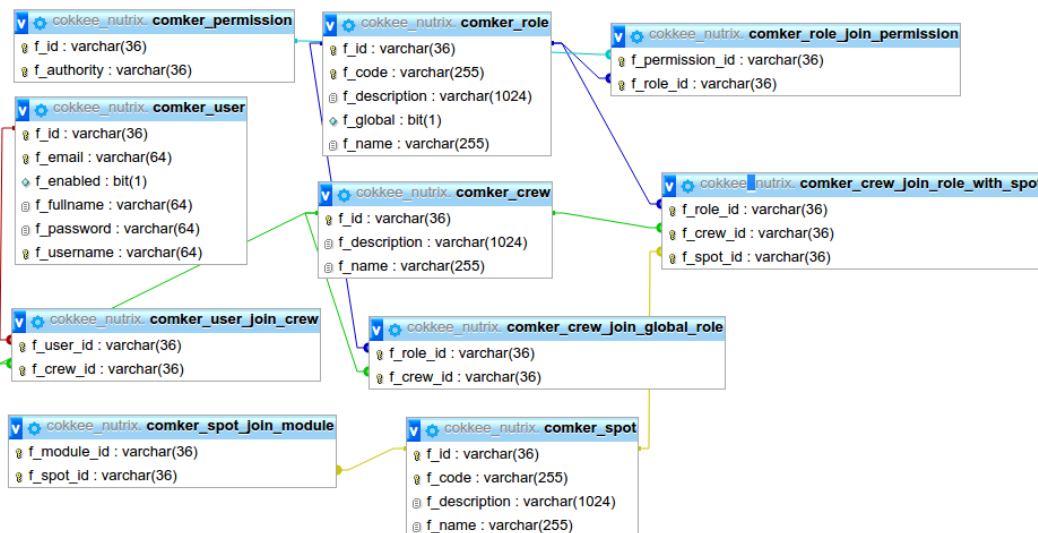
8 Cơ sở dữ liệu

- Cơ sở dữ liệu về thông tin thực phẩm, chất dinh dưỡng và thực đơn.



Hình 2.17: Cơ sở dữ liệu về dinh dưỡng

- Cơ sở dữ liệu về quản lý hệ thống.



Hình 2.18: Cơ sở dữ liệu về quản lý hệ thống

9 Mô tả dữ liệu

- Bảng nutrix_ingredient

Tên trường	Kiểu dữ liệu	Ghi chú
f_id	varchar(36)	Mã của thực phẩm (Primary Key)
f_description	varchar(255)	Mô tả thực phẩm
f_label	varchar(255)	Tên thực phẩm
f_code	varchar(64)	Tên phân biệt duy nhất của thực phẩm
f_db	varchar(16)	Nguồn của cơ sở dữ liệu
f_id	varchar(16)	Mã tham chiếu

Bảng 2.3: Cấu trúc thông tin dữ liệu của Ingredient

- Bảng nutrix_nutrient

Tên trường	Kiểu dữ liệu	Ghi chú
f_id	varchar(36)	Mã của chất dinh dưỡng (Primary Key)
f_code	varchar(64)	Tên phân biệt duy nhất của chất dinh dưỡng
f_description	varchar(255)	Mô tả thông tin chất dinh dưỡng
f_name	varchar(255)	Tên chất dinh dưỡng

f_num_dec	tinyint(4)	Chữ số làm tròn phần thập phân của giá trị dinh dưỡng
f_origin_flag	int(11)	
f_unit_type	varchar(4)	Đơn vị đo của chất dinh dưỡng
f_ref_db	varchar(16)	Nguồn của cơ sở dữ liệu
f_ref_id	varchar(16)	Mã tham chiếu

Bảng 2.4: Cấu trúc thông tin dữ liệu của nutrient

- Bảng nutrix_ingredient_join_nutrient

Tên trường	Kiểu dữ liệu	Ghi chú
f_amount	double	Giá trị của chất dinh dưỡng có trong thực phẩm
f_ingredient_id	varchar(36)	Mã của thực phẩm (Primary Key)
f_nutrient_id	varchar(36)	Mã của chất dinh dưỡng (Primary Key)
f_num_data_pts	int(11)	Giá trị trung bình
f_nutr_max	double	Giá trị lớn nhất
f_nutr_min	double	Giá trị nhỏ nhất
f_std_error	double	Độ lệch chuẩn

Bảng 2.5: Cấu trúc thông tin dữ liệu của bảng ingredient_join_nutrient

- Bảng nutrix_recipe

Tên trường	Kiểu dữ liệu	Ghi chú
f_id	varchar(36)	Mã của thực đơn(Primary Key)
f_description	varchar(255)	Mô tả thực đơn
f_name	varchar(255)	Tên thực đơn
f_per_serving	int(11)	Số lượng người dùng thực đơn

Bảng 2.6: Cấu trúc thông tin dữ liệu của bảng nutrix_recipe

- Bảng nutrix_recipe_join_ingredient

Tên trường	Kiểu dữ liệu	Ghi chú
f_amount	double	Khối lượng thực phẩm dùng trong thực đơn
f_measure_unit	varchar(255)	Đơn vị tính của thực phẩm
f_ingredient_id	varchar(36)	Mã của thực phẩm
f_recipe_id	varchar(36)	Mã của thực đơn

Bảng 2.7: Cấu trúc thông tin dữ liệu bảng nutrix_recipe_join_ingredient

Dữ liệu về thông tin thành phần dinh dưỡng của một loại thực phẩm được tổng hợp từ nguồn dữ liệu: cơ sở dữ liệu lấy từ website của Bộ Nông nghiệp Hoa Kỳ- USDA Food Composition Database, version 26, năm 2013 (xem[13]). Ví dụ mô tả cấu trúc thông tin dữ liệu thành phần dinh dưỡng của thực phẩm thịt heo nạc (pork) của Hoa Kỳ. Chú thích:

- Tên các thành phần dinh dưỡng được viết bằng tiếng anh.
- Đối với các thành phần còn thiếu số liệu (chưa phân tích, không có số liệu tham khảo, ký hiệu bằng dấu gạch ngang (-)).
- Đối với các thành phần không có trong thực phẩm, số liệu được ghi là 0.
- Thành phần dinh dưỡng tính trong 100g phần ăn được.

Bảng thành phần dinh dưỡng của thịt heo từ nguồn của Bộ Nông Nghiệp Hoa Kỳ -USDA.

Nutrient	Unit	1 value per 100g
Proximates		
Water	g	60.6
Energy	kcal	247
Protein	g	15.3
Total lipid (fat)	g	19.87
Carbohydrate, by difference	g	0.73
Fiber, total dietary	g	0

Sugars, total	g	0
Minerals		
Calcium, Ca	mg	11
Iron, Fe	mg	0.77
Magnesium, Mg	mg	14
Phosphorus, P	mg	139
Potassium, K	mg	281
Sodium, Na	mg	907
Zinc, Zn	mg	2.03
Vitamins		
Vitamin C, total ascorbic acid	mg	0
Thiamin	mg	0.523
Riboflavin	mg	0.157
Niacin	mg	3.9
Vitamin B-6	mg	0.27
Folate, DFE	$\hat{A}\mu\text{g}$	5
Vitamin B-12	$\hat{A}\mu\text{g}$	0.93
Vitamin A, RAE	$\hat{A}\mu\text{g}$	0
Vitamin A, IU	IU	0
Vitamin E (alpha-tocopherol)	mg	0.26
Vitamin D (D2 + D3)	$\hat{A}\mu\text{g}$	1.4
Vitamin D	IU	56
VitaminK(phyloquinone)	$\hat{A}\mu\text{g}$	0.3

Lipids		
Fatty acids, total saturated	g	6.839
Fatty acids, total monounsaturated	g	9.732
Fatty acids, total monounsaturated	g	2.107
Cholesterol	mg	59
Other		
Caffeine	mg	0

Bảng 2.8: Bảng mô tả cấu trúc thông tin dữ liệu thành phần dinh dưỡng của thực phẩm thịt heo nạc (pork) của Hoa Kỳ.

Luận văn này phát triển công cụ chuyển đổi dữ liệu từ Access được lấy từ nguồn (<http://www.ars.usda.gov/Services/docs.htm?docid=24936>) sang cấu trúc cơ sở dữ liệu MySQL.

10 Thiết kế giao diện

1. Màn hình mô tả chức năng tìm kiếm thông tin chất dinh dưỡng có trong 1 loại thực phẩm do người dùng chọn.

Nutrition

[List ingredient](#)

- Butter, salted
- Butter, whipped, with salt
- Butter oil, anhydrous
- Cheese, blue
- Cheese, brick
- Cheese, brie
- Cheese, camembert
- Cheese, caraway
- Cheese, cheddar
- Cheese, cheshire
- Cheese, colby
- Cheese, cottage, creamed, large or small

Hình 2.19: Chức năng tìm kiếm thông tin.

2. Màn hình mô tả kết quả tìm thông tin thực phẩm.

Nutrition Fact

Name: Butter, salt

Name	Value	Unit
Protein	0.86	g
Ash	2.11	g
Energy	717	kcal
Water	15.87	g
Caffeine	0	mg
Sugars, total	0.06	g

Nutrition Information

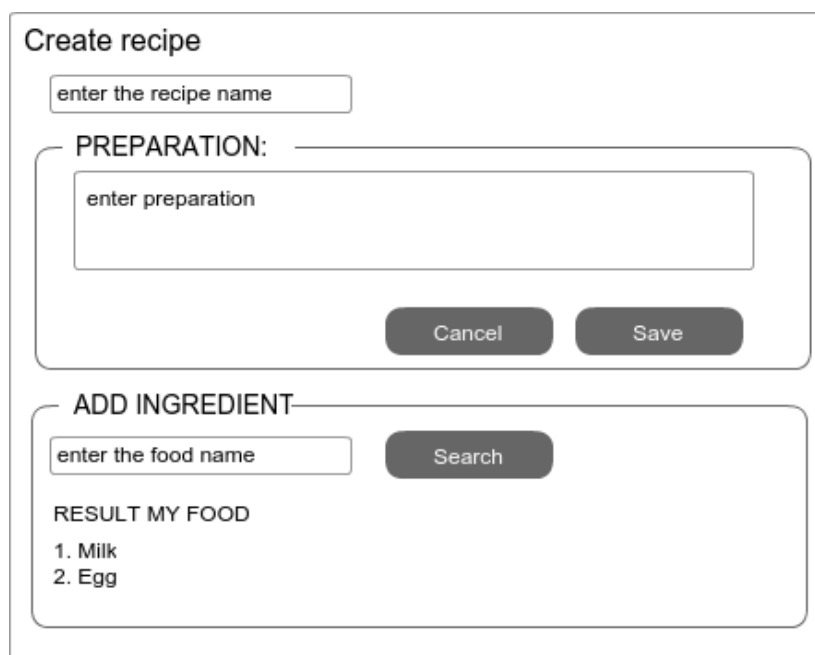
Amount per 100 grams

Protein information

Calorie information

Hình 2.20: Kết quả tìm kiếm thực phẩm.

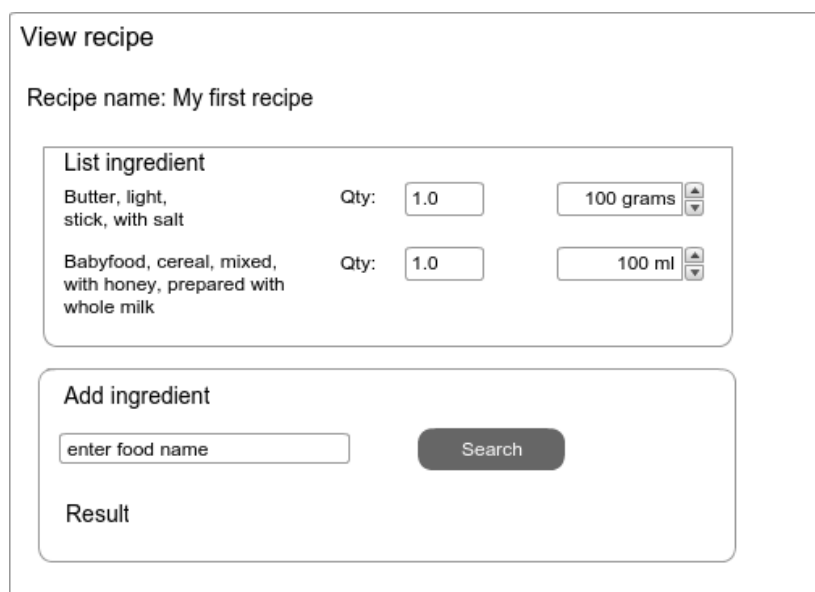
3. Màn hình mô tả chức năng tạo mới thực đơn.



The 'Create recipe' form is divided into two main sections. The top section, titled 'PREPARATION:', contains a text input field labeled 'enter the recipe name' and a larger text area labeled 'enter preparation'. Below these are 'Cancel' and 'Save' buttons. The bottom section, titled 'ADD INGREDIENT', contains a text input field labeled 'enter the food name' and a 'Search' button. Below the search field, under the heading 'RESULT MY FOOD', is a list of results: '1. Milk' and '2. Egg'.

Hình 2.21: Chức năng tạo mới thực đơn.

4. Màn hình xem chi tiết thực đơn .



The 'View recipe' form displays the details of a recipe. At the top, it shows 'Recipe name: My first recipe'. Below this is a section titled 'List ingredient' which contains two items: 'Butter, light, stick, with salt' with a quantity of '1.0' and a unit of '100 grams', and 'Babyfood, cereal, mixed, with honey, prepared with whole milk' with a quantity of '1.0' and a unit of '100 ml'. Each item has a quantity input field and a unit dropdown menu. Below the list is a section titled 'Add ingredient' with a text input field labeled 'enter food name' and a 'Search' button. At the bottom, there is a section titled 'Result'.

Hình 2.22: Chi tiết thực đơn.

Kết luận

Chương 2 trình bày về phân tích và thiết kế hệ thống. Trong chương này nêu lên được những vấn đề sau:

- Mô tả các chức năng của hệ thống
- Yêu cầu về đặc điểm kỹ thuật của hệ thống.
- Mô tả chức năng dưới dạng User Story.
- Mô tả và cài đặt Acceptance Test.
- Các sơ đồ Usecase, Sequence Diagram và Class Diagram.

Dựa trên những đặc tả, phân tích và thiết kế hệ thống đã trình bày, chương 3 của luận văn sẽ tập trung vào việc xây dựng và phát triển hệ thống.

CHƯƠNG 3

KẾT QUẢ ĐẠT ĐƯỢC

1 Các giao diện

1.1 Giao diện trang bìa

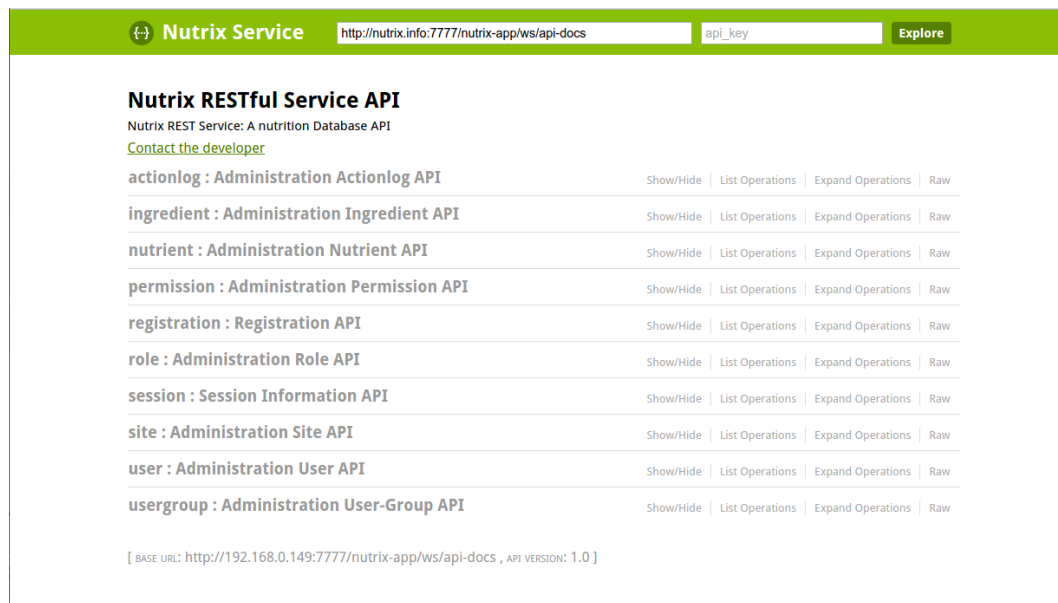


Hình 3.1: Giao diện trang bìa

Là một giao diện trang web đơn giản cung cấp thông tin và kết nối đến các ứng dụng được trình bày trong luận văn. Trang web được đặt tại địa chỉ <http://nutrix.info>.

1.2 Giao diện nutrix_webservice

Hình 3.2 là giao diện truy cập vào cơ sở dữ liệu thông qua Restful Service. Mỗi tài nguyên (ví dụ nutrient, ingredient,...) được cung cấp một bộ phương thức HTTP (GET, PUT, POST, DELETE) để người dùng thao tác dữ liệu thông qua truy vấn HTTP.



Hình 3.2: Giao diện nutrix web service

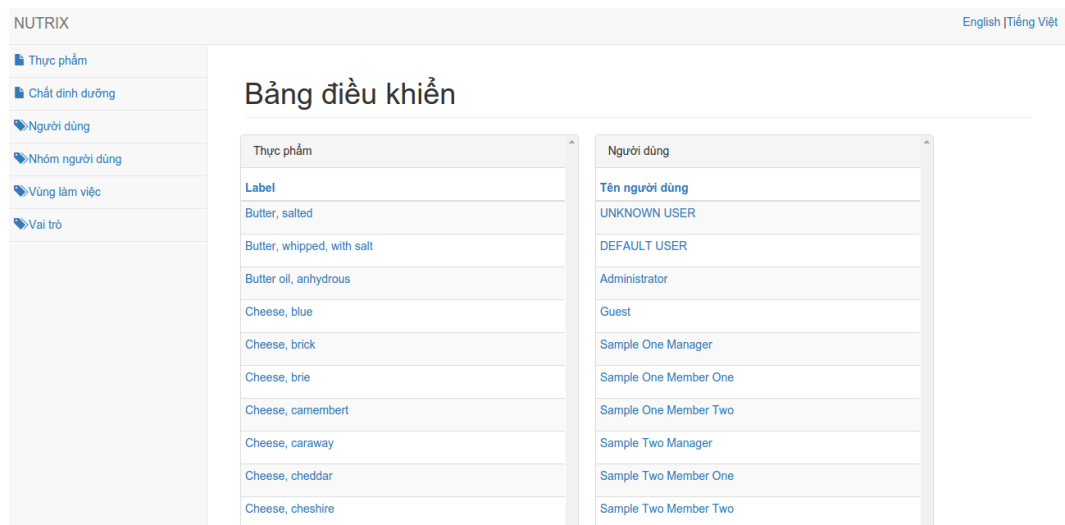
Hình 3.3 là giao diện truy cập vào tài nguyên nutrient với các phương thức HTTP (GET, PUT, POST, DELETE).

nutrient : Administration Nutrient API		Show/Hide	List Operations	Expand Operations	Raw
POST	/ws/nutrix/adm/nutrient	Add a new nutrient to the database			
GET	/ws/nutrix/adm/nutrient	List all of nutrients			
GET	/ws/nutrix/adm/nutrient/getbycode/{code}	Get the detail of nutrient content by code			
PUT	/ws/nutrix/adm/nutrient/{id}	Update an existing nutrient			
GET	/ws/nutrix/adm/nutrient/{id}	Get a nutrient by ID			
DELETE	/ws/nutrix/adm/nutrient/{id}	Delete an existing nutrient			

Hình 3.3: Giao diện khai thác dữ liệu cho nutrient

1.3 Giao diện nutrix_ngadmin

Nutrix_ngadmin là giao diện dành cho người quản lý hệ thống với các chức năng quản lý (thêm, sửa, xóa) các thực phẩm, chất dinh dưỡng...và các chức năng quản lý tài khoản người dùng.



Hình 3.4: Giao diện trang quản lý hệ thống

Hình 3.5 là giao diện xem thông tin thực phẩm với các chức năng người quản lý có thể chọn là thêm, sửa, xóa thực phẩm.

NUTRIX

English | Tiếng Việt

Thực phẩm

Chất dinh dưỡng

Người dùng

Nhóm người dùng

Vùng làm việc

Vai trò

Xem tất cả thực phẩm

Lọc tìm

Q

X

+ Tạo mới

Mã	Tên thực phẩm	Thao tác
	Butter, salted	Xem Sửa Xóa
BUTTER,WHIPPED,WITH SALT	Butter, whipped, with salt	Xem Sửa Xóa
BUTTER OIL,ANHYDROUS	Butter oil, anhydrous	Xem Sửa Xóa
CHEESE,BLUE	Cheese, blue	Xem Sửa Xóa
CHEESE,BRICK	Cheese, brick	Xem Sửa Xóa
CHEESE,BRIE	Cheese, brie	Xem Sửa Xóa
CHEESE,CAMEMBERT	Cheese, camembert	Xem Sửa Xóa
CHEESE,CARAWAY	Cheese, caraway	Xem Sửa Xóa
CHEESE,CHEDDAR	Cheese, cheddar	Xem Sửa Xóa
CHEESE,CHESHIRE	Cheese, cheshire	Xem Sửa Xóa
CHEESE,COLBY	Cheese, colby	Xem Sửa Xóa
CHEESE,COTTAGE,CRMD,LRG OR SML CURD	Cheese, cottage, creamed, large or small curd	Xem Sửa Xóa
CHEESE,COTTAGE,CRMD,W/FRUIT	Cheese, cottage, creamed, with fruit	Xem Sửa Xóa

Hình 3.5: Giao diện trang xem thông tin các thực phẩm

Hình 3.6 là giao diện xem thông tin chất dinh dưỡng như tên, kí hiệu và đơn vị tính của chất dinh dưỡng.

NUTRIX

English | Tiếng Việt

Thực phẩm

Chất dinh dưỡng

Người dùng

Nhóm người dùng

Vùng làm việc

Vai trò

Xem tất cả chất dinh dưỡng

Lọc tìm

Q

X

Mã	Tên chất dinh dưỡng	Đơn vị	Thao tác
PROCNT	Protein	g	Xem
FAT	Total lipid (fat)	g	Xem
CHOCDF	Carbohydrate, by difference	g	Xem
ASH	Ash	g	Xem
ENERC_KCAL	Energy	kcal	Xem
STARCH	Starch	g	Xem
SUCS	Sucrose	g	Xem
GLUS	Glucose (dextrose)	g	Xem
FRUS	Fructose	g	Xem
LACS	Lactose	g	Xem
MALS	Maltose	g	Xem
ALC	Alcohol, ethyl	g	Xem
WATER	Water	g	Xem

Hình 3.6: Giao diện trang xem thông tin chất dinh dưỡng

62

Hình 3.7 là giao diện chỉnh sửa thông tin của thực phẩm như thay đổi tên, mô tả, và thay đổi hàm lượng chất dinh dưỡng có trong thực phẩm.

Sửa thực phẩm "Cheese, blue Cheese, blue"

Tên thực phẩm:

Mã:

Danh sách chất dinh dưỡng có trong thực phẩm

Tên thực phẩm	Giá trị	Giá trị tối thiểu	Giá trị tối đa	Giá trị trung bình	Sai số trung bình
Protein	21.4	-	-	19	0.339

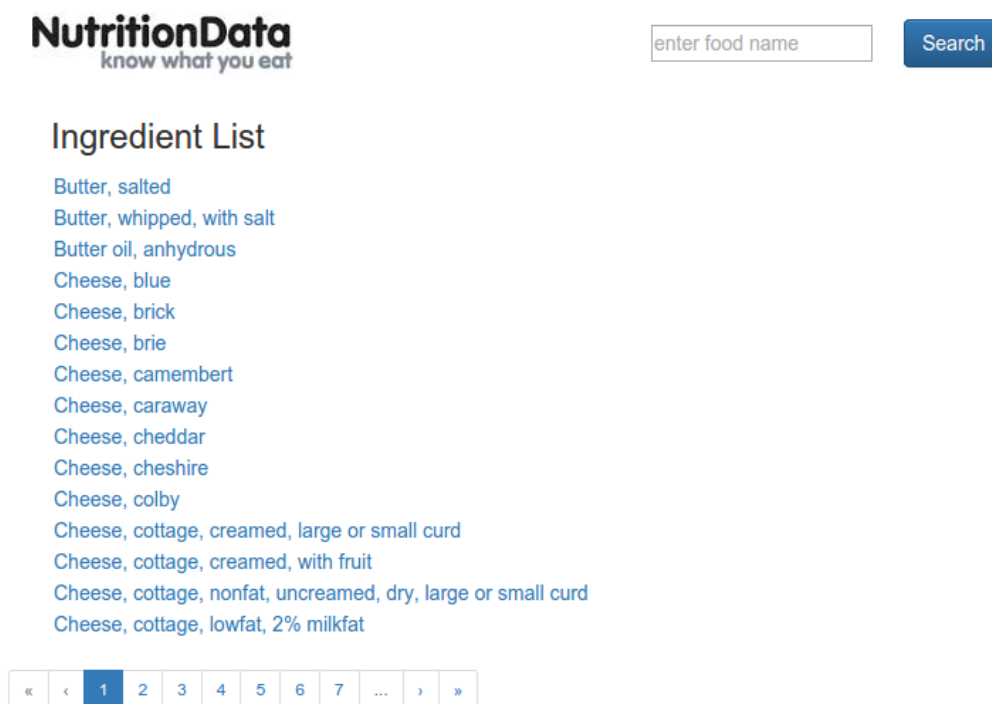
Sửa Xóa

Hình 3.7: Giao diện trang sửa thông tin thực phẩm

1.4 Giao diện nutrix_webapp

Nutrix_webapp là phần khai thác dành cho người sử dụng hệ thống. Người dùng có thể tìm kiếm, xem thông tin thực phẩm, giá trị chất dinh dưỡng trong thực phẩm và tạo mới thực đơn.

Hình 3.8 là giao diện tìm kiếm thông tin thực phẩm với đầy đủ hàm lượng dinh dưỡng có trong thực phẩm.



Hình 3.8: Giao diện trang quản lý người dùng

Hình 2.2 là giao diện xem chi tiết thực phẩm với hàm lượng dinh dưỡng có trong thực phẩm. Các hàm lượng được tính trong 100g thực phẩm ăn được.

Nutrition Fact

Name: Cheese, brick

Nutrient	Value 100g	Unit
Protein	23.24	g
Total lipid (fat)	29.68	g
Carbohydrate, by difference	2.79	g
Ash	3.18	g
Energy	371	kcal
Starch		g
Sucrose		g
Glucose (dextrose)		g
Fructose		g
Lactose		g
Maltose		g
Alcohol, ethyl	0	g
Water	41.11	g

Hình 3.9: Giao diện trang xem chi tiết thực phẩm

Hình 3.10 là giao diện tạo mới thực đơn theo nhu cầu của người dùng khi chọn thực phẩm và số lượng người sử dụng thực đơn.

Recipe list

Recipe name

Chicken Soup for the soul - Servers 3 people

Cupcake milk - Servers 5 people

Chocolate milk - Servers 7 people

Black Bean and Salsa Soup - Servers 3 people

Shaking Beef - Servers 6 people

Shaker Lemon Pie - Servers 5 people

Kiwi Salsa - Servers 4 people

Almond Udon Salad - Servers 2 people

Apple Pancake - Servers 3 people

Polish Noodles - Servers 2 people

Create recipe

Recipe Nutrition Calculator

Recipe name

Number of servings

serves people

add material

calories (kcal)	carbs (g)	fat (g)	protein (g)	sodium (mg)	sugar (g)	cholesterol (mg)	action
--------------------	--------------	------------	----------------	----------------	--------------	---------------------	--------

+

Save recipe

Delete

Summary

Total:	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Per Serving:	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Hình 3.10: Giao diện trang tạo mới thực đơn

Hình 3.11 là giao diện chỉnh sửa thực đơn với các chỉnh sửa thêm hoặc xóa thực phẩm. Lúc này các thông số về chất dinh dưỡng có trong thực đơn cũng được tính toán lại.

Recipe list

Recipe name

Chicken Soup for the soul - Servers 3 people

Cupcake milk - Servers 5 people

Chocolate milk - Servers 7 people

Black Bean and Salsa Soup - Servers 3 people

Shaking Beef - Servers 6 people

Shaker Lemon Pie - Servers 5 people

Kiwi Salsa - Servers 4 people

Almond Udon Salad - Servers 2 people

Apple Pancake - Servers 3 people

Polish Noodles - Servers 2 people

Create recipe

Recipe Nutrition Calculator

Recipe name

Chocolate milk

Number of servings

serves people

add material

	calories (kcal)	carbs (g)	fat (g)	protein (g)	sodium (mg)	sugar (g)	cholesterol (mg)	action
Butter, salted(2x100 grams)	1,434.00	0.12	162.22	1.72	1,286.00	0.12	430.00	-
Butter, whipped, with salt(3x100 grams)	2,151.00	0.18	243.33	2.55	1,977.00	0.18	657.00	-
Cheese, blue(33x100 grams)	11,649.00	77.22	948.42	706.20	37,818.00	16.50	2,475.00	-

+

Save recipe

Delete

Summary

Total:	15,234.00	77.52	1,353.97	710.47	41,081.00	16.80	3,562.00
Per Serving:	2,176.29	11.07	193.42	101.50	5,868.71	2.40	508.86

Hình 3.11: Giao diện trang chỉnh sửa thực đơn

Hình 3.12 là giao diện thêm mới thực phẩm trong thực đơn với các lựa chọn thực phẩm và số lượng thực phẩm sử dụng trong thực đơn.

Recipe list

Recipe name

Chicken Soup for the soul - Servers 3 people

Cupcake milk - Servers 5 people

Chocolate milk - Servers 7 people

Black Bean and Salsa Soup - Servers 3 people

Shaking Beef - Servers 6 people

Shaker Lemon Pie - Servers 5 people

Kiwi Salsa - Servers 4 people

Almond Udon Salad - Servers 2 people

Apple Pancake - Servers 3 people

Polish Noodles - Servers 2 people

Recipe Nutrition Calculator

Recipe name

Number of servings

serves people

add material

calories (kcal)	carbs (g)	fat (g)	protein (g)	sodium (mg)	sugar (g)	cholesterol (mg)	action
-----------------	-----------	---------	-------------	-------------	-----------	------------------	--------

Add Material To Recipe

Search our food database by name:

enter food name

Matching Foods:

Butter, salted

Butter, whipped, with salt

Butter oil, anhydrous

Cheese, blue

Cheese, brick

Cheese, brie

(choose one ingredient, please!)

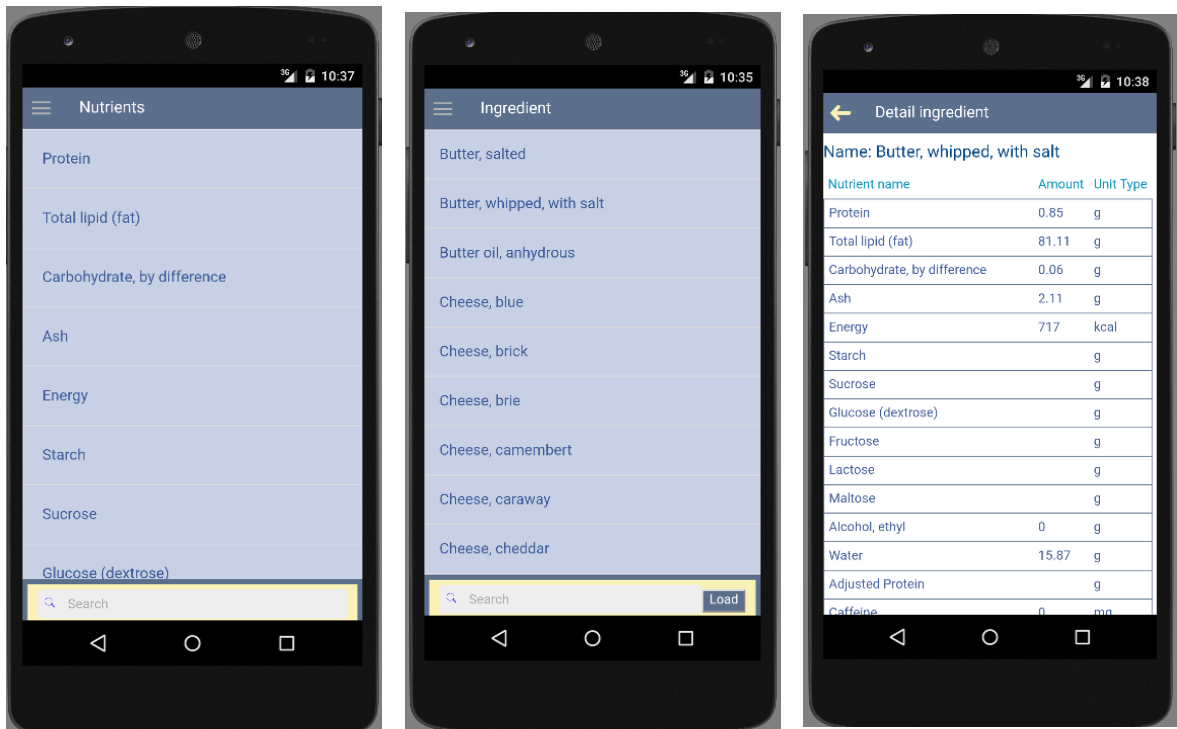
How much?

serving of (Choose one, please) ▼

Hình 3.12: Giao diện trang thêm mới thực phẩm trong thực đơn

1.5 Giao diện nutrix_mobiapp

Là giao diện trên mobile sử dụng Ionic framework. Với Ionic, có thể chạy trên Android, iOS, Winphone nhưng với luận văn này, vì điều kiện còn hạn chế nên ứng dụng này chỉ chạy trên Android.



Hình 3.13: Các giao diện xem thông tin thực phẩm, chất dinh dưỡng và xem chi tiết thông tin thực phẩm.

Các hình trên là giao diện hiển thị xem danh sách thực phẩm, danh sách chất dinh dưỡng và xem chi tiết thực phẩm trên thiết bị di động chạy bằng Android.

2 Cài đặt và cấu hình hệ thống

Phần này trình bày các bước cài đặt và cấu hình hệ thống. Hệ thống (có mã dự án là nutrix - từ đây để tiện theo dõi sẽ gọi đơn giản là hệ thống nutrix hoặc nutrix) bao gồm dịch vụ chạy trên máy chủ, ứng dụng web hoạt động

trên trình duyệt desktop/laptop, ứng dụng chạy trên điện thoại cài đặt hệ điều hành Android. Để vận hành nutrix, một số phần mềm bên ngoài (tất cả đều là mã nguồn mở) cần phải được cài đặt, bao gồm Java, MySQL Server, Tomcat Server (dùng cho dịch vụ web); Trình chủ Apache Server (dùng cho ứng dụng web chạy trên trình duyệt).

2.1 Cài đặt MySQL và Apache Server

Để thuận tiện cho việc triển khai và bảo trì, chúng ta có thể sử dụng các bản đóng gói sẵn như WAMP, LAMP, XAMPP. Trong luận văn này, chúng ta sẽ cài đặt hệ thống trên Linux Ubuntu, do đó sẽ cài đặt bản XAMPP.

- Bước 1: Mở terminal và tải về gói cài đặt XAMPP 1.8.3

Với 32-bit:

```
wget http://sourceforge.net/projects/xampp/files/XAMPP%20Linux/1.8.3/xampp-linux-1.8.3-2-installer.run/download
```

Với 64-bit:

```
wget http://sourceforge.net/projects/xampp/files/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-2-installer.run/download
```

- Bước 2: Thay đổi quyền thực thi cho XAMPP thành root và chạy XAMPP bằng lệnh sau:

```
sudo chmod +x xampp-linux-x64-1.8.3-2-installer.run
```

```
sudo ./xampp-linux-x64-1.8.3-2-installer.run
```

Khi đó giao diện XAMPP xuất hiện, nhấn Next để tiếp tục cài đặt.

Giao diện sau khi hoàn thành cài đặt:



Hình 3.14: Giao diện hoàn thành cài đặt XAMPP

Khởi động lại dịch vụ XAMPP: `sudo /opt/lampp/lampp start`

Dừng các dịch vụ XAMPP: `sudo /opt/lampp/lampp stop`

Để mở dịch vụ XAMPP, gõ vào thanh địa chỉ trình duyệt của bạn:

`http: //localhost/xampp/`

2.2 Java

Các bước cài đặt đều được thực hiện trên Terminal.

- Bước 1: Cài đặt JRE/JDK

Update các gói (package): `sudo apt-get update`

Kiểm tra java đã được cài đặt trong máy: `java -version`

Nếu xuất hiện "The program java can be found in the following packages" thì Java chưa được cài đặt, thực hiện lệnh sau để cài đặt:

```
sudo apt-get install default-jre
```

Lệnh này để cài đặt JRE (Java Runtime Environment), nếu bạn cần JDK (Java Development Kit) thực hiện lệnh sau:

```
sudo apt-get install default-jdk
```

- **Bước 2: Cài đặt OpenJDK 7**

Để cài đặt OpenJDK 7, thực hiện lệnh sau:

```
sudo apt-get install openjdk-7-jre
```

Nếu bạn cần JDK thì thực hiện lệnh sau:

```
sudo apt-get install openjdk-7-jdk
```

- **Bước 3: Cài đặt Oracle JDK**

Thực hiện các lệnh sau để cài đặt Oracle JDK

```
sudo apt-get install python-software-properties
```

```
sudo add-apt-repository ppa:webupd8team/java
```

```
sudo apt-get update
```

Sau đó, tùy thuộc vào phiên bản bạn lựa chọn mà thực hiện lệnh sau:

- Oracle JDK 6 : `sudo apt-get install oracle-java6-installer`
- Oracle JDK 7 : `sudo apt-get install oracle-java7-installer`
- Oracle JDK 8 : `sudo apt-get install oracle-java8-installer`

- Bước 4: Kiểm tra các bản cài đặt Java Để kiểm tra các bản cài đặt Java, sử dụng lệnh sau:

`sudo update-alternatives --config java`

Nếu có 2 phiên bản cài đặt, trên terminal xuất hiện:

```
There are 2 choices for the alternative java (providing /usr/bin/java).

Selection    Path                                          Priority    Status
-----
* 0          /usr/lib/jvm/java-7-oracle/jre/bin/java      1062       auto mode
 1          /usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java 1061       manual mode
 2          /usr/lib/jvm/java-7-oracle/jre/bin/java      1062       manual mode

Press enter to keep the current choice[*], or type selection number:
```

Tại đây, bạn có thể chọn số để sử dụng mặc định. Điều này cũng được sử dụng với trình biên dịch Java (javac)

`sudo update-alternatives --config javac`

- Bước 5: Thiết lập các biến môi trường "JAVA_HOME" Để thiết lập các biến môi trường, đầu tiên phải tìm ra đường dẫn cài đặt java, sử dụng lệnh sau:

`sudo update-alternatives --config java`

```
There are 2 choices for the alternative java (providing /usr/bin/java).

Selection    Path                                          Priority    Status
-----
* 0          /usr/lib/jvm/java-7-oracle/jre/bin/java      1062       auto mode
 1          /usr/lib/jvm/java-6-openjdk-amd64/jre/bin/java 1061       manual mode
 2          /usr/lib/jvm/java-7-oracle/jre/bin/java      1062       manual mode

Press enter to keep the current choice[*], or type selection number:
```

Đường dẫn cài đặt cho mỗi tùy chọn trên là:

1. `/usr/lib/jvm/java-7-oracle`

2. /usr/lib/jvm/java-6-openjdk-amd64

3. /usr/lib/jvm/java-7-oracle

Sao chép đường dẫn từ cài đặt của bạn và sau đó chỉnh sửa các tập tin /
etc / environment:

```
sudo nano /etc/environment
```

Trong file này, thêm dòng sau (thay thế YOUR_PATH bởi đường dẫn):

```
JAVA_HOME="YOUR_PATH"
```

Sau khi thiết lập biến môi trường xong, tải lại file:

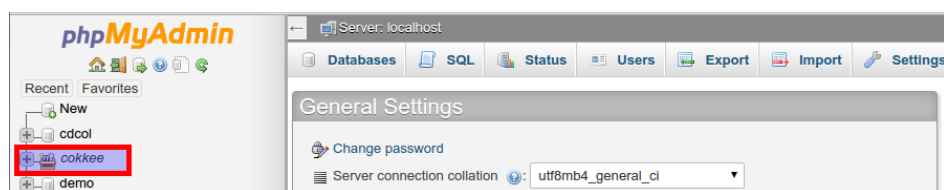
```
source /etc/environment
```

Kiểm tra nó bằng cách thực hiện:

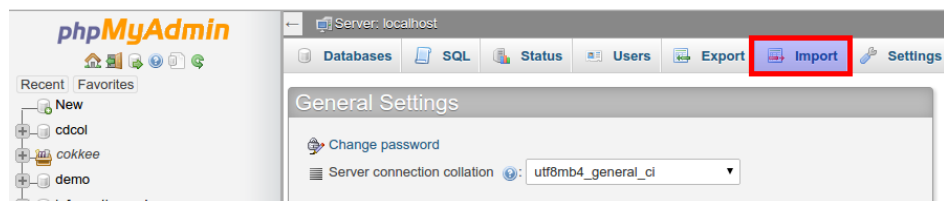
```
echo $JAVA_HOME
```

2.3 Import cơ sở dữ liệu vào MYSQL sử dụng phpMyAdmin

Trong phpMyAdmin, chọn cơ sở dữ liệu của bạn từ danh sách bên trái.



Chọn "Import".



Chọn vào "Choose file".

Importing into the database "cokkee_nutrix"

File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **[format].[compression]**. Example: **.sql.zip**

Browse your computer: **Choose File** No file chosen (Max: 512MiB)

Character set of the file: utf-8

Partial Import:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to import large files, / transactions.)

Skip this number of queries (for SQL) or lines (for other formats), starting from the first one: 0

Format:

SQL

Format-Specific Options:

SQL compatibility mode: NONE

☒ Do not use AUTO_INCREMENT for zero values

Go

Đến nơi chứa tập tin SQL của bạn và chọn "Open".

File cần import hiển thị như hình

Importing into the database "cokkee_nutrix"

File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **[format].[compression]**. Example: **.sql.zip**

Browse your computer: **Choose File** nutrix_ingredient.sql (Max: 512MiB)

Character set of the file: utf-8

Partial Import:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good transactions.)

Skip this number of queries (for SQL) or lines (for other formats), starting from the first one: 0

Format:

SQL

Format-Specific Options:

SQL compatibility mode: NONE

☒ Do not use AUTO_INCREMENT for zero values

Go

Nhấp vào nút "Go" phía dưới cùng của trang. Tùy thuộc vào kích thước dữ liệu, việc import dữ liệu có thể mất một vài phút.

2.4 Tomcat Server và nutrix-service

2.4.1 Tomcat Server

- Bước 1: Mở Terminal và tải về bản cài đặt Tomcat Server.

```
wget http://mirrors.ibiblio.org/apache/tomcat/tomcat-8/v8.0.9/bin/apache-tomcat-8.0.9.tar.gz
```

Sau khi tải về hoàn tất, giải nén các file:

```
tar xvfz apache-tomcat-8.0.9.tar.gz
```

Sau đó, di chuyển vào một vị trí thích hợp:

```
mv apache-tomcat-8.0.9 /opt/tomcat
```

- Bước 2: Cấu hình .bashrc.

Thiết lập các biến môi trường trong .bashrc

```
vim /.bashrc
```

Thêm thông tin này vào cuối file:

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
```

```
export CATALINA_HOME=/opt/tomcat
```

Bạn cần lưu và thoát khỏi .bashrc, sau đó thực hiện lệnh sau để thấy được sự thay đổi:

```
./bashrc
```

Để kích hoạt Tomcat, sử dụng lệnh sau:

```
$CATALINA_HOME/bin/startup.sh
```

Kết quả trả lại:

Using CATALINA_BASE: /opt/tomcat

Using CATALINA_HOME: /opt/tomcat

Using CATALINA_TMPDIR: /opt/tomcat/temp

Using JRE_HOME: /usr/lib/jvm/java-7-openjdk-amd64/

Using CLASSPATH: /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar

Tomcat started.

2.4.2 Nutrix-service

- Bước 1: Tại file target trong nutrix-service (nutrix-service/target), chọn tập tin nutrix-service.war.
- Bước 2: Copy tập tin nutrix-service.war và chuyển vào thư mục webapps của Tomcat (apache-tomcat-7.0.61/webapps).
- Bước 3: Trong thư mục /nutrix-service/src/main/webapp/WEB-INF tìm đến file jdbc.properties và thay đổi thông số username, password để kết nối đến MySQL.

3 Mã nguồn chương trình

Toàn bộ mã nguồn chương trình được chia sẻ công khai trên kho mã nguồn mở Github, tại địa chỉ <https://github.com/nutrix-info/> để tất cả mọi người đều có thể tải về (bằng công cụ git) chỉnh sửa, biên dịch, thử nghiệm, mở rộng.

KẾT LUẬN VÀ KIẾN NGHỊ

Qua thời gian được học tập và tìm hiểu các công nghệ mới, sau thời gian gần sáu tháng để hoàn thành luận văn, em đã xây dựng được ứng dụng để tra cứu và xây dựng thực đơn dinh dưỡng dựa trên mô hình Web Service và One-page Application. Thông qua luận văn này, em đã hiểu thêm về các dịch vụ Web (Web Service), và cũng được học thêm về những công nghệ mới, xây dựng một Web Service bằng Java, sử dụng Spring Framework và phát triển ứng dụng phía client trên web và thiết bị di động với AngularJS, Ionic framework... Với thời gian và kỹ năng có hạn, ứng dụng còn thiếu sót rất nhiều tính năng nên hướng phát triển và nghiên cứu thêm là:

1. Có thể phát triển thêm các chức năng khác như lưu trữ thực đơn, tính năng lượng cần thiết tùy theo mỗi người...
2. Thu thập chính xác các dữ liệu về dinh dưỡng tại Việt Nam để làm đa dạng hơn cơ sở dữ liệu.
3. Phát triển hệ thống hướng đa nền tảng, đa thiết bị.
4. Bảo mật dữ liệu cho website.

Với kinh nghiệm tiếp cận công nghệ mới còn hạn chế nên kết quả của ứng dụng này còn rất nhiều thiếu sót. Rất mong thầy cô và các bạn đóng góp ý kiến để ứng dụng được hoàn thiện thêm. Cuối cùng, em xin chân thành cảm ơn toàn thể giảng viên khoa công nghệ thông tin Đại học Sư Phạm Đà Nẵng,

và đặc biệt là giảng viên hướng dẫn Nguyễn Trần Quốc Vinh đã nhiệt tình giúp đỡ em hoàn thành đề tài này.

Xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

- [1] AngularJS. Conceptual overview. URL <https://docs.angularjs.org/guide/concepts>.
- [2] Scott Ambler + Associates. User stories: An agile introduction. URL <http://www.agilemodeling.com/artifacts/userStory.htm>.
- [3] Dan Bunea. How tdd improves development speed and is very cost effective, September 22, 2005. URL <http://danbunea.blogspot.com/2005/09/how-tdd-improves-development-speed-and.html>.
- [4] William C.Wake. *Refactoring Workbook*. Addison-Wesley, 2004.
- [5] Viện dinh dưỡng.
- [6] Viện dinh dưỡng. Chiến lược quốc gia về dinh dưỡng giai đoạn 2011-2020 và tầm nhìn đến năm 2030. *Viện dinh dưỡng*, pages 1–2, 02-2012.
- [7] drifty. The ionic book, 2014. URL <http://ionicframework.com/docs/guide/>.
- [8] FAO. New developments in nutrition education utilising computer tech-

- nology, 2014. URL <http://www.fao.org/docrep/w3733e/w3733e07.htm>.
- [9] Happyworm. The future of web apps – single page applications, Monday, August 23rd, 2010. URL <http://happyworm.com/blog/2010/08/23/the-future-of-web-apps-single-page-applications/>.
- [10] javaworld.com. Xfire, 2014. URL <http://www.javaworld.com/article/2071759/soa/xfire--the-easy-and-simple-way-to-develop-web-services.html>.
- [11] Jersey. Restlet, 2014. URL <https://jersey.java.net/>.
- [12] Quỹ nhi đồng Liên hợp quốc-UNICEF tại Việt Nam. Phòng chống “các rối loạn do thiếu i-ốt” ở việt nam, 11-2007. URL http://www.unicef.org/vietnam/vi/media_16839.html.
- [13] United States Department of Agriculture. Usda national nutrient database for standard reference. URL 2014.
- [14] Ruby on Rails insights for developers. Single page application done right, Monday, August 23rd, 2010. URL <http://railsadventures.wordpress.com/2014/05/04/single-page-application-done-right/>.
- [15] Oracle. *The Java EE 6 Tutorial - Filtering Requests and Responses*. Oracle, .

- [16] Oracle. *The Java EE 6Tutorial - Developing RESTfulWeb Services with JAX-RS*. Oracle, .
- [17] Oracle. *The Java EE 6Tutorial*. Oracle, .
- [18] Oracle. *Developing Applications For the Java EE 6 Platform*. Oracle, .
- [19] Oracle. Jdbc-oracle, 2014. URL <http://www.oracle.com/technetwork/java/javase/jdbc/index.html>.
- [20] Oracle. Servlet engines, 2014. URL http://docs.oracle.com/cd/A91202_01/901_doc/java.901/a90213/introduction.htm.
- [21] Oracle. Annotations, 2014. URL <http://docs.oracle.com/javase/tutorial/java/annotations/>.
- [22] David Orenstein. How to application programming interface. *Computerworld*, October, 2000.
- [23] PGS. TS. Hà Thị Anh Đào PGS. TS. Nguyễn Công Khẩn. *Bảng thành phần thực phẩm Việt Nam*. Nhà xuất bản Y học.
- [24] resteasy. Resteasy, 2014. URL <http://resteasy.jboss.org/>.
- [25] Science, 2014. URL sciencedaily.com.
- [26] James Shore. *The Art of Agile Development*. 2010.
- [27] Pathfinder Solutions. The pros and cons of tdd, June 11, 2012. URL <http://www.pathfindersolns.com/archives/1684>.

- [28] tutorialspoint. Http-request, 2014. URL http://www.tutorialspoint.com/http/http_requests.htm.
- [29] tutorialspoint. Http-response, 2014. URL http://www.tutorialspoint.com/http/http_responses.htm.
- [30] Liên Hợp Quốc tại Việt Nam. Dinh dưỡng bà mẹ và trẻ em ở việt nam, 2013. URL <http://www.un.org.vn/vi/feature-articles-press-centre-submenu-252/339-dinh-dung-ba-m-va-tr-em--vit-nam.html>.
- [31] UncleBob's. The three laws of tdd. URL <http://butunclebob.com/ArticleS.UncleBob.TheThreeRulesOfTdd>.
- [32] USDA. About the database, 2014. URL <http://ndb.nal.usda.gov/>.
- [33] versionone. Test-first programming. URL http://www.versionone.com/Agile101/Test-First_Programming.asp.
- [34] Wikipedia. Apache cxf, 2014. URL http://en.wikipedia.org/wiki/Apache_CXF.
- [35] Wikipedia. Android, 2014. URL [http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system)).
- [36] Wikipedia. Celtix, 2014. URL <http://en.wikipedia.org/wiki/Celtix>.

- [37] Wikipedia. Hypertext transfer protocol, 2014. URL http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol.
- [38] Wikipedia. Representational state transfer, 2014. URL http://en.wikipedia.org/wiki/Representational_state_transfer.
- [39] Wikipedia. Restlet, 2014. URL <http://en.wikipedia.org/wiki/Restlet>.
- [40] Wikipedia. Single-page application, 2014. URL http://en.wikipedia.org/wiki/Single-page_application.
- [41] Wikipedia. Web container, 2014. URL http://en.wikipedia.org/wiki/Web_container.
- [42] ThS. BS Đào thị Yến Phi. Đại cương về dinh dưỡng. pages 1–2.

PHỤ LỤC

1 Mã nguồn chương trình