**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



─────── \* ───────



**BÁO CÁO**

Web thế hệ mới

Đề tài: Xây dựng Web thương mại điện tử thông minh

*Sinh viên thực hiện:*

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và Tên** | **MSSV** | **Lớp** |
| 1 | Nguyễn Công Hưng | 20131945 | CNTT1.02 |
| 2 | Trần Trí Dũng | 20130704 | CNTT1.02 |
| 3 | Phạm Mạnh Hùng | 20131907 | CNTT1.02 |

*Giáo Viên hướng dẫn*: **TS. Phạm Huy Hoàng**

HÀ NỘI- 5/2017

**Mục Lục**

[Chương 1 Tổng quan đề tài 3](#_Toc482788339)

[Chương 2 Xây dựng mô hình web 4](#_Toc482788340)

[2.1 Mô hình tổng quan 4](#_Toc482788341)

[2.2 Mô hình lập trình 4](#_Toc482788342)

[Chương 3 Giải thuật áp dụng trong mô hình 6](#_Toc482788343)

[3.1 Giới thiệu giải thuật áp dụng 6](#_Toc482788344)

[3.2 Áp dụng trong bài toán cụ thể 6](#_Toc482788345)

[Chương 4 Triển khai và demo 7](#_Toc482788346)

[4.1 Ngôn ngữ và nền tảng để viết 7](#_Toc482788347)

[4.2 Xây dựng cơ sở dữ liệu 7](#_Toc482788348)

[4.3 Xử lý dữ liệu nhận được 9](#_Toc482788349)

[4.3.1 Các lớp đối tượng 9](#_Toc482788350)

[4.3.2 Các lớp giải thuật 15](#_Toc482788351)

[4.3.3 Các lớp giao diện 21](#_Toc482788352)

[4.3.4 Kết nối cơ sở dữ liệu 25](#_Toc482788353)

[Chương 5 Đánh giá tổng kết 27](#_Toc482788354)

[5.1 Ưu điểm 27](#_Toc482788355)

[5.2 Nhược điểm 27](#_Toc482788356)

[5.3 Mở rộng 27](#_Toc482788357)

# Chương 1 Tổng quan đề tài

Webste từ lâu đã là lĩnh vực quan trọng trong cuộc sống nói chung và công nghệ thông tin nói riêng.

Kể từ khi thế hệ web 1.0 ra đời rồi đến 2.0, việc quảng bá thông tin, giới thiệu sản phẩm, dịch vụ đến các hệ thống quản lý hay ứng dụng tích hợp web đều được nâng lên và đạt đến mức độ hoàn hảo cho người sử dụng

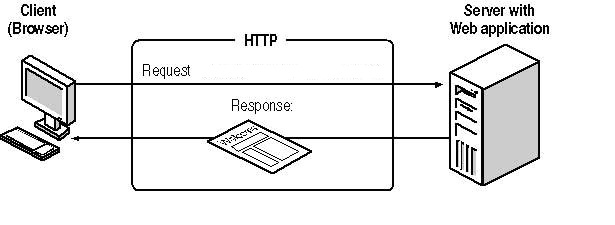
Ngày nay thông tin càng ngày càng nhiều, dữ liệu lưu trữ từ các hệ cơ sở dữ liệu quan hệ lẫn phi quan hệ càng ngày càng lớn dẫn tới tình trạng thừa dữ liệu. Điều đó ảnh hưởng rất lớn đến các website mang tính định hướng cho người sử dụng, người sử dụng sẽ tốn nhiều thời gian và công sức hơn để có được các thông tin cần và có ích với mình trên các website

Giải quyết vấn đề này nhiều nhà nghiên cứu, các trường đại học, viên nghiên cứu trên thế giới hay chính cha đẻ của các thế hệ web 1.0, 2.0 đã đưa ra một khái niệm về thế hệ web 3.0

Website 3.0 về mô hình chung vẫn đang được nghiên cứu nhưng đã có nhiều đề suất được đưa ra như đưa automatic intelligent vào hệ thống web nhằm giúp con người giảm công sức tìm kiếm cũng như giúp máy tính hiểu được dữ liệu. Các ý tưởng đó có khá nhiều như web semantic, web phân tích dữ liệu (chứng khoán, thời tiết.v.v.v).

# Chương 2 Xây dựng mô hình web

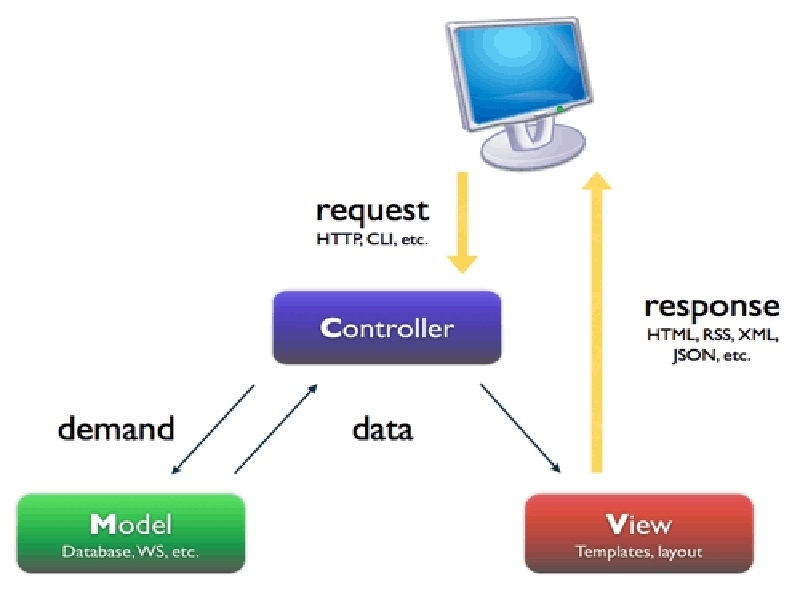
## 2.1 Mô hình tổng quan



Client sẽ làm nhiệm vụ chính là giao tiếp trực tiếp với người sử dụng

Server làm nhiệm vụ thu nhận request từ phía client site gửi đến sau đó sẽ lưu trữ , xử lý và phản hồi lại bên phía client

## 2.2 Mô hình lập trình



Mô hình lập trình được nhóm em áp dụng là mô hình MVC.

Trong đó view là các giao diện web browser giúp giao tiếp với con người.

Controller ở bên server sẽ nhận các request từ phía client sau đó sẽ xử lý bao gồm: làm việc với cơ sở dữ liệu, tính toán dựa trên các giải thuật được cài đặt và response cho view bên phía client site.

Model là các dữ liệu trong cơ sở dữ liệu hoặc các object.

# Chương 3 Giải thuật áp dụng trong mô hình

## 3.1 Giới thiệu giải thuật áp dụng

Như phần mở đầu của báo cáo chúng em có nói đến sự khác nhau rõ rệt nhất giữa thế hệ web 2.0 và 3.0 là ở khả năng hiểu được của máy tính đối với dữ liệu mà con người sử dụng.

Có rất nhiều cách để giúp máy tính hiểu được dữ liệu mà con người sử dụng như giúp máy tính hiểu được ý nghĩa của từ (ngữ nghĩa của từ), tính toán các khả năng xảy ra (dự đoán trước), tìm ra các quy luật mà con người hay làm, v.v.v.

Trong bài tập lớn này em đã chọn và sử dụng giải thuật Navi Bayes để giải quyết vấn đề về dự báo trước xu hướng dựa trên dữ liệu thu thập được từ người sử dụng.

## 3.2 Áp dụng trong bài toán cụ thể

Bài toán nhóm đặt ra:

*Thiết kế xây dựng web bán hàng (thuộc kiểu thương mại điện tử) bao gồm client và server.*

*Sử dụng giải thuật Navi Bayes để dự đoán trước những sản phầm tiềm năng đối với từng khách hàng dựa trên thông tin cá nhân cũng như các dữ liệu hóa đơn từ nhiều khách hàng đã mua trước.*

Ghi chú: Phần trình bày chi tiết về giải thuật navi bayes cũng như cách áp dụng chúng em sẽ viết trong phần **4.3.2 Các lớp giải thuật**

# Chương 4 Triển khai và demo

## 4.1 Ngôn ngữ và nền tảng để viết

Đầu tiên là phía client làm việc với người sử dụng chúng em sẽ sử dụng thuần túy HTML (JSP), CSS, JavaScript để lập trình giao diện web browser

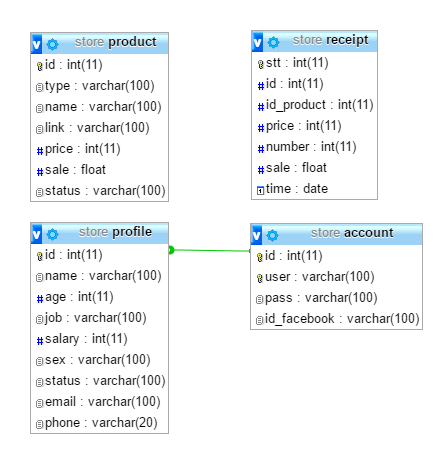
Tiếp theo là phía server chúng em sẽ sử dụng java (servlet) để viết phần controller

Cuối cùng là cơ sở dữ liệu sẽ được xây dựng bằng hệ quản trị cơ sở dữ liệu MySQL

Nền tảng để viết toàn bộ chương trình là jdk 8 và môi trường IDE NetBeans 8.1

## 4.2 Xây dựng cơ sở dữ liệu

Để bài toán được đơn giản và tập trung nhiều vào giải thuật chúng em đã thiết kế cơ sở dữ liệu tập trung chủ yếu vào sản phẩm trong cửa hàng cũng như thông tin cá nhân khách hàng



Như hình chúng em ghi lại từ cơ sở dữ liệu : Mỗi khách hàng sẽ có 1 tài khoản (account) dùng để đăng nhập vào hệ thống cửa hàng.

Tài khoản này sẽ được người sử dụng tự lập thông qua đăng ký với hệ thống bằng 1 trong 2 cách: đăng ký normal và đăng ký qua tài khoản facebook

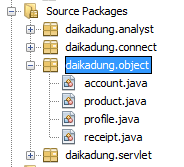
Sau khi đăng ký xong hệ thống sẽ nhận được các thông tin cá nhân của khách hàng và lưu trữ tại cở sở dữ liệu để sau này dùng để phân tích

Trong cơ sở dữ liệu còn có 1 bảng là product dùng để lưu trữ thông tin sản phẩm

Ngoài ra em đã dùng 1 bảng là receipt dùng để lưu trữ hóa đơn đã từng mua của các khách hàng trước đây để thuận tiên cho việc xử lý giải thuật

## 4.3 Xử lý dữ liệu nhận được

### 4.3.1Các lớp đối tượng



*public class product {*

*private int id;*

*private String name;*

*private String type;*

*public String getType() {*

*return type;*

*}*

*public void setType(String type) {*

*this.type = type;*

*}*

*private String link;*

*private int price;*

*private float sale;*

*public void setId(int id) {*

*this.id = id;*

*}*

*public void setName(String name) {*

*this.name = name;*

*}*

*public void setLink(String link) {*

*this.link = link;*

*}*

*public void setPrice(int price) {*

*this.price = price;*

*}*

*public void setSale(float sale) {*

*this.sale = sale;*

*}*

*public int getId() {*

*return id;*

*}*

*public String getName() {*

*return name;*

*}*

*public String getLink() {*

*return link;*

*}*

*public int getPrice() {*

*return price;*

*}*

*public float getSale() {*

*return sale;*

*}*

*}*

*public class profile {*

*private String name;*

*private int age;*

*private String job;*

*private int salary;*

*private String sex;*

*private String status;*

*private String email;*

*private String phone;*

*public String getName() {*

*return name;*

*}*

*public int getAge() {*

*return age;*

*}*

*public String getJob() {*

*return job;*

*}*

*public int getSalary() {*

*return salary;*

*}*

*public String getSex() {*

*return sex;*

*}*

*public String getStatus() {*

*return status;*

*}*

*public String getEmail() {*

*return email;*

*}*

*public String getPhone() {*

*return phone;*

*}*

*public void setName(String name) {*

*this.name = name;*

*}*

*public void setAge(int age) {*

*this.age = age;*

*}*

*public void setJob(String job) {*

*this.job = job;*

*}*

*public void setSalary(int salary) {*

*this.salary = salary;*

*}*

*public void setSex(String sex) {*

*this.sex = sex;*

*}*

*public void setStatus(String status) {*

*this.status = status;*

*}*

*public void setEmail(String email) {*

*this.email = email;*

*}*

*public void setPhone(String phone) {*

*this.phone = phone;*

*}*

*}*

*public class receipt {*

*private String name;*

*private String link;*

*private int number;*

*private int price;*

*private float sale;*

*private String time;*

*public int getNumber() {*

*return number;*

*}*

*public int getPrice() {*

*return price;*

*}*

*public float getSale() {*

*return sale;*

*}*

*public String getTime() {*

*return time;*

*}*

*public void setName(String name) {*

*this.name = name;*

*}*

*public void setLink(String link) {*

*this.link = link;*

*}*

*public String getName() {*

*return name;*

*}*

*public String getLink() {*

*return link;*

*}*

*public void setNumber(int number) {*

*this.number = number;*

*}*

*public void setPrice(int price) {*

*this.price = price;*

*}*

*public void setSale(float sale) {*

*this.sale = sale;*

*}*

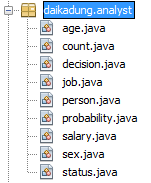
*public void setTime(String time) {*

*this.time = time;*

*}*

*}*

### 4.3.2 Các lớp giải thuật



Mô tả: mỗi khi một người dùng truy cập vào hệ thống và chọn chức năng gợi ý sản phẩm, 1 request từ client sẽ gửi đến server

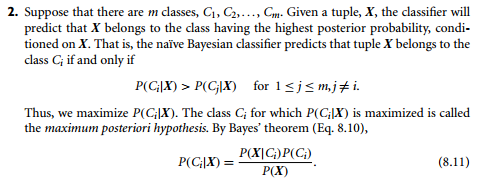
Server sẽ dựa vào các class analyst để tính toán các sản phẩm phù hợp với khách hàng đó

Quá trình tính toán chia làm 3 bước:

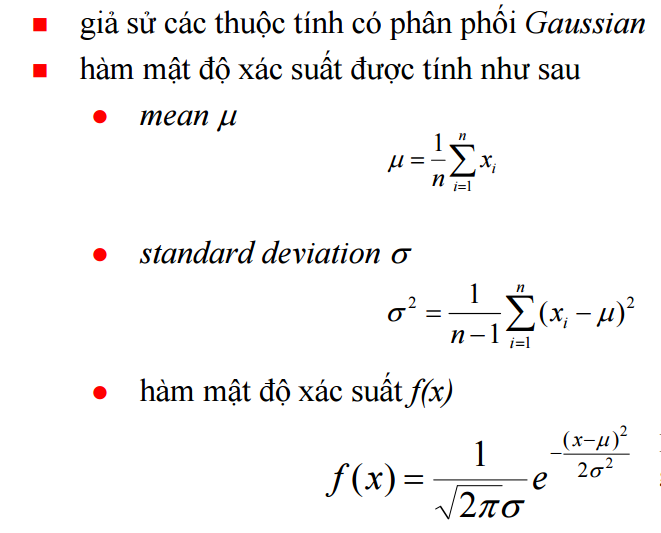
* Lấy tất cả dữ liệu trong bảng profile
* Tính toán xác suất của mỗi thành phần bằng cách đếm số lượng từ cơ sở dữ liệu rồi chia cho tổng
* Tính xác suất quyết định bằng cách nhân tất cả các xác suất thành phần, sau đó so sánh giữa xác suất mua và không mua để đưa ra quyết định

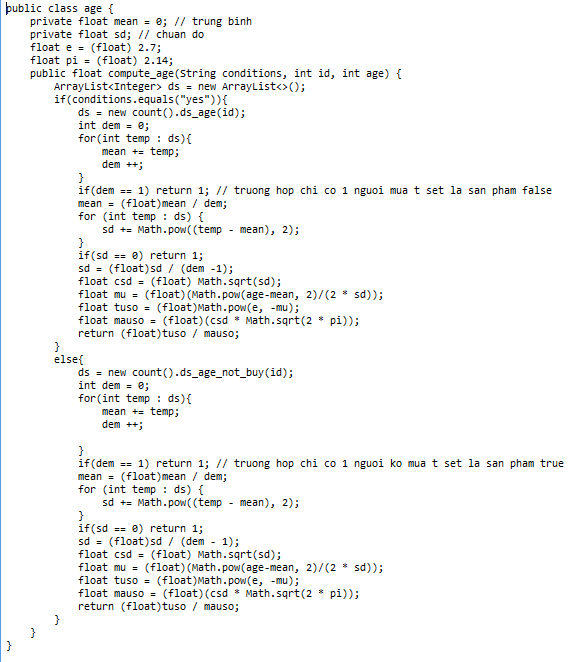


Với các lớp có kiểu dữ liệu rời rạc như job, status, gender e sẽ dùng công thức tính xác suất có điều kiện:

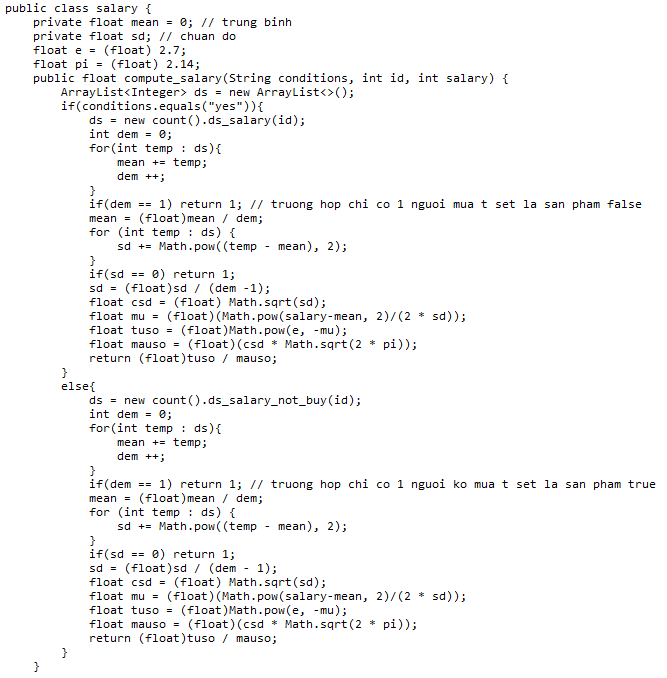
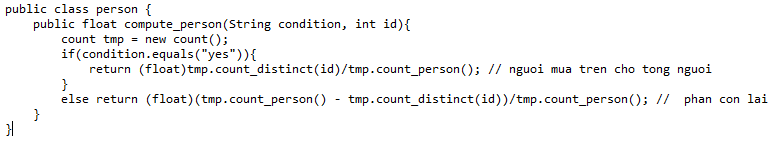


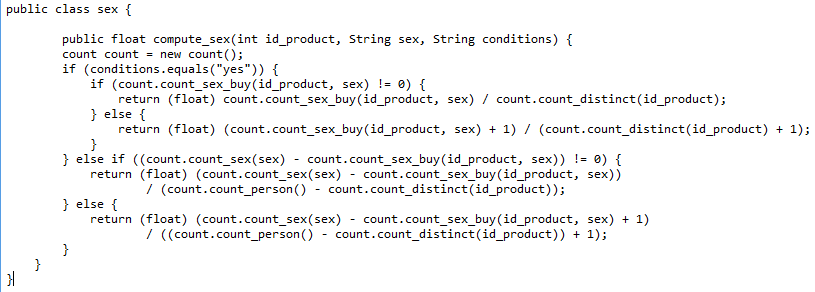
Với các lớp như có kiểu dữ liệu là là số và liên tục như age, salary em sẽ dùng công thức:

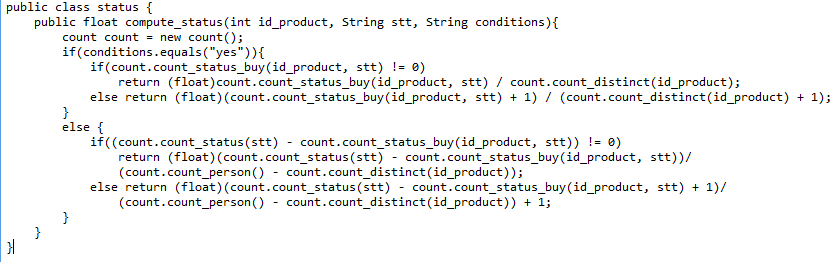




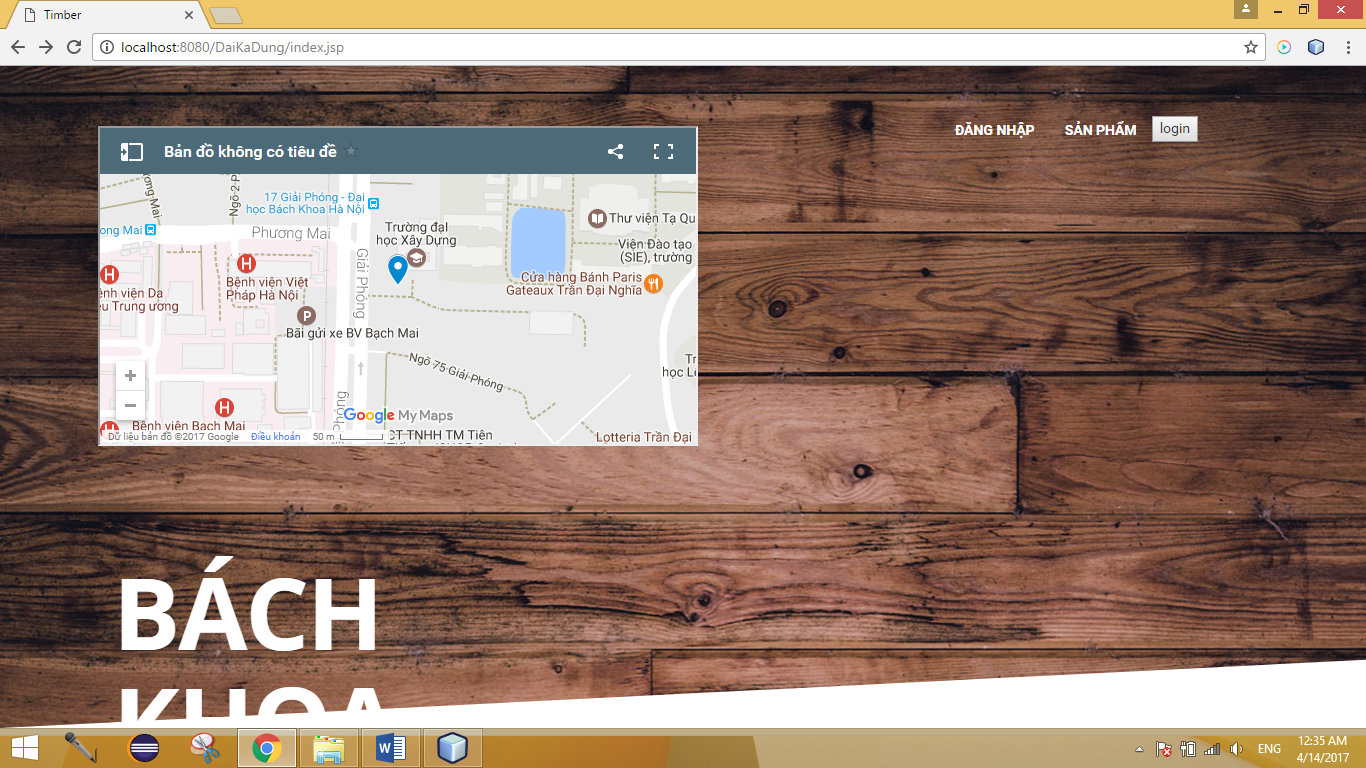


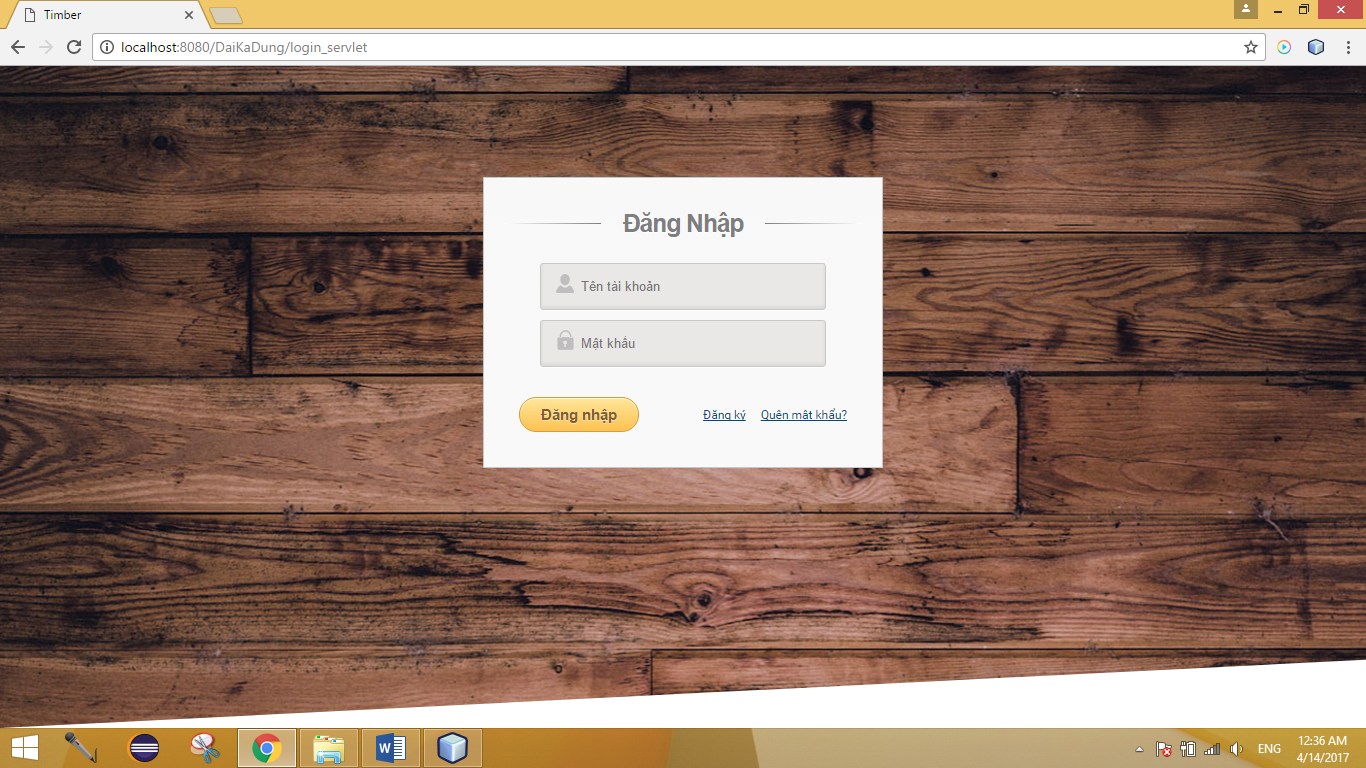


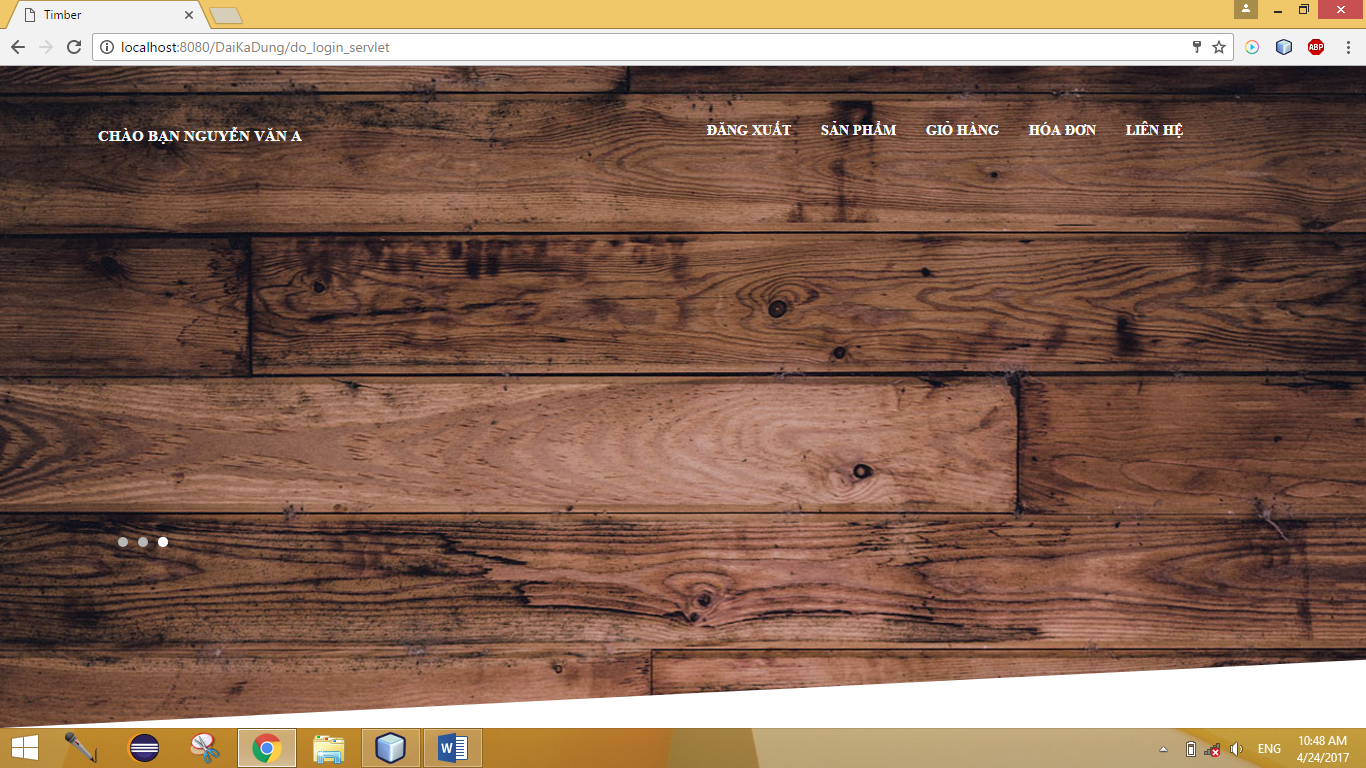


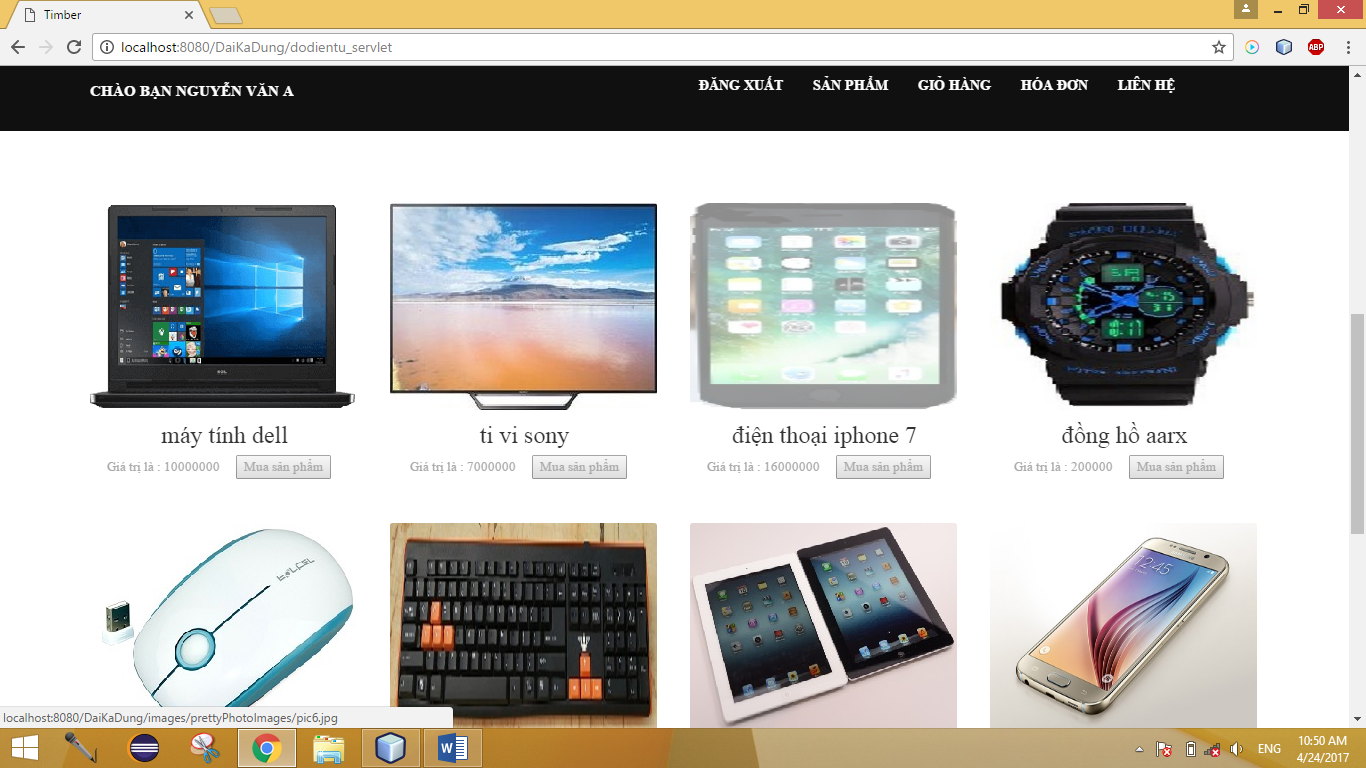


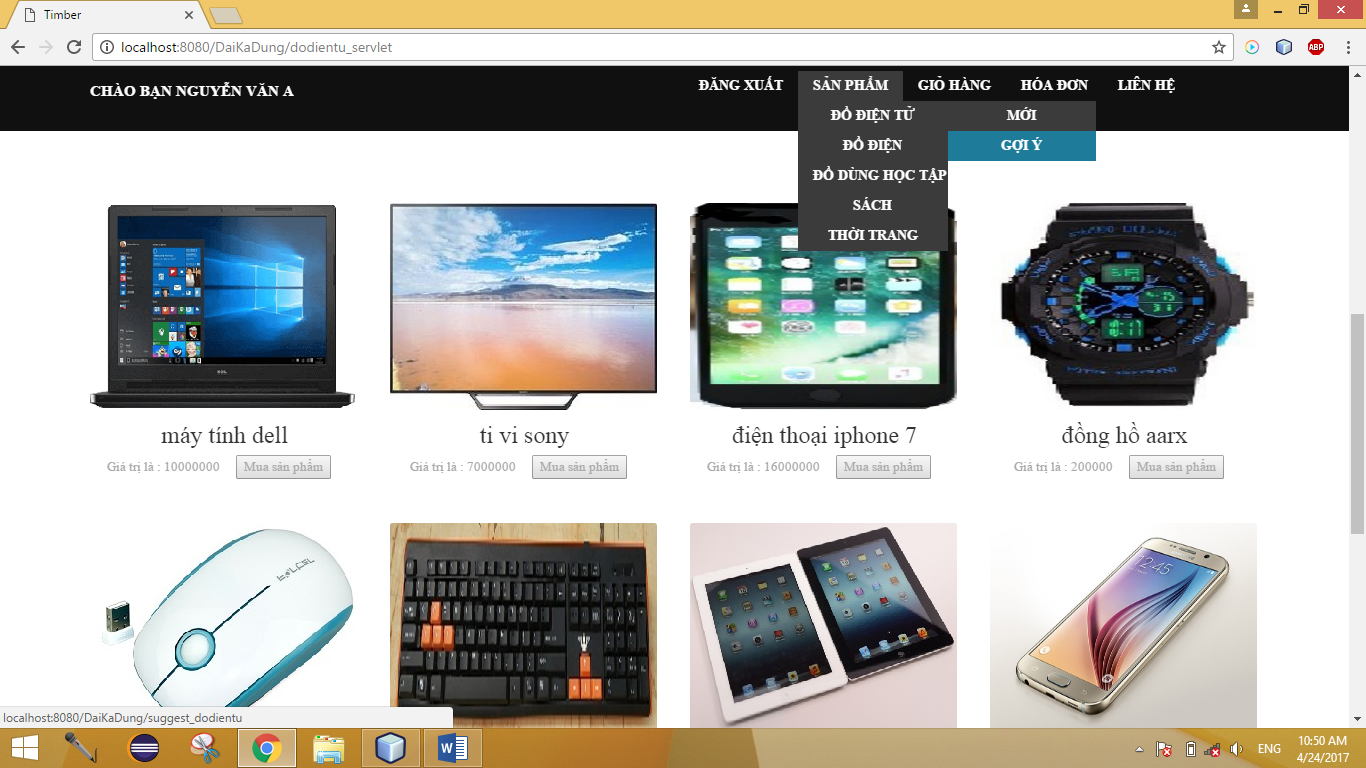
### 4.3.3 Các lớp giao diện

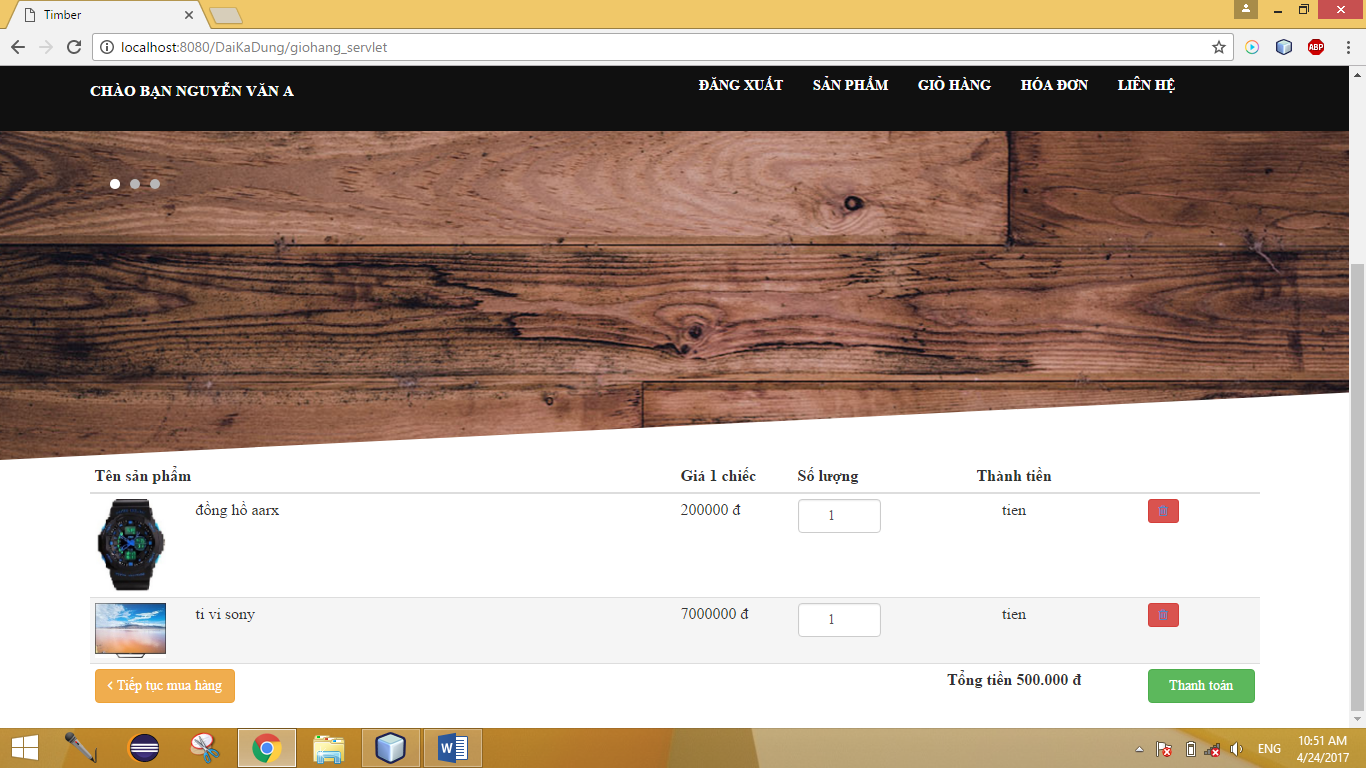


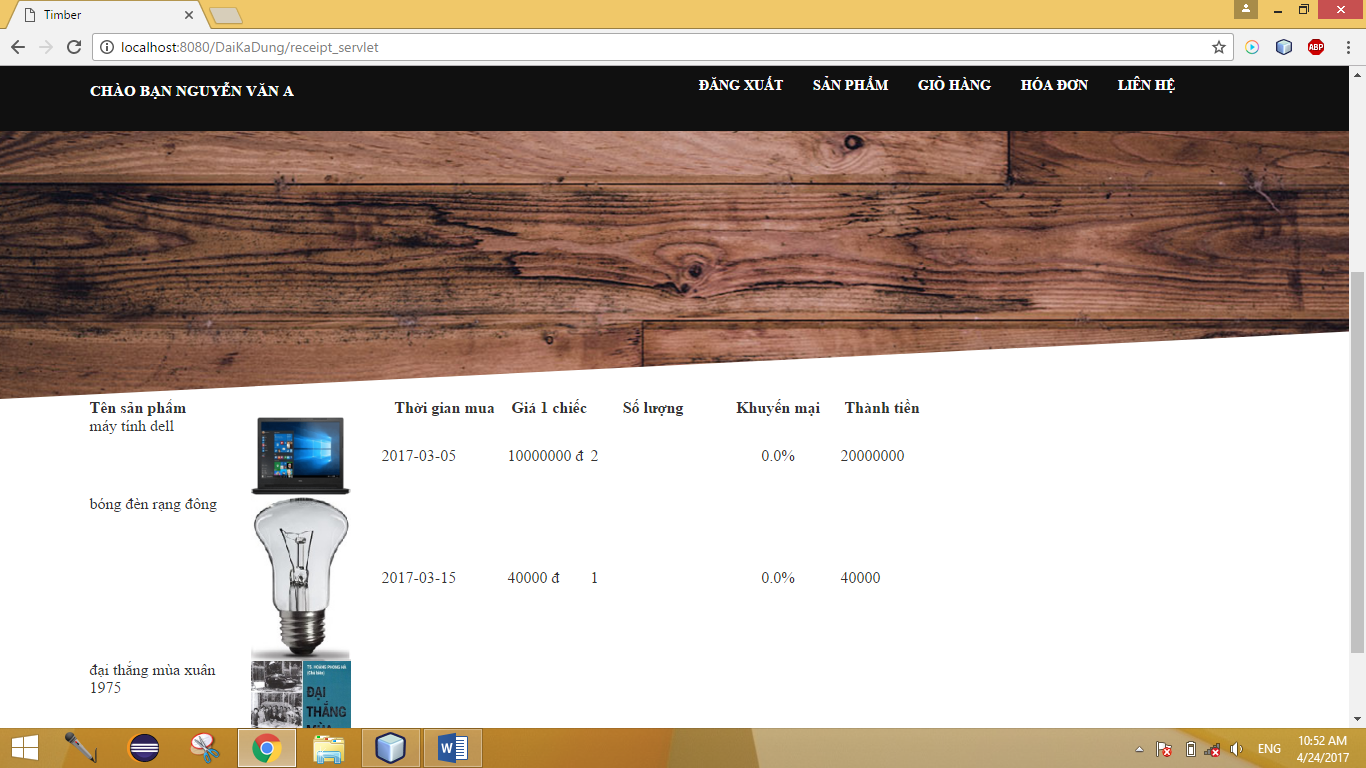












### 4.3.4 Kết nối cơ sở dữ liệu

Để kết nối cơ sở dữ liệu nhóm chúng em sử dụng api JDBC mysql để kết nối.



# Chương 5 Đánh giá tổng kết

## 5.1 Ưu điểm

Sử dụng được giải thuật khai phá dữ liệu (Navi Bayes) trong việc phân tích dữ liệu nhằm tìm ra các sản phầm tiềm năng.

Giúp máy tính hiểu được dữ liệu người dùng

## 5.2 Nhược điểm

Giao diện chưa bắt mắt

Thiết kế cơ sở dữ liệu chưa phong phú mới chủ yếu tập trung phục vụ giải thuật

## 5.3 Mở rộng

Sau chương trình này em muốn lồng ghép thêm các giải thuật khai phá dữ liệu như frequent pattern hoặc mạng bayes tin cậy

Sau khi đưa ra một sản phẩm khách hàng có thể mua, áp dụng frequent pattern để đưa tiếp chuỗi sản phầm liên quan đến sản phẩm vừa gợi ý

VD: Khi tìm ra một khách hàng có nghề nghiệp là kĩ sư máy tính, dựa vào navi bayes chương trình sẽ đưa ra gợi ý mua máy tính cho khách hàng.

Sau đó dựa vào frequent pattern, chương trình sẽ tiếp tục đưa ra chuỗi sản phẩm gợi ý dựa vào sản phẩm máy tính như bàn phím, chuột, v.v.v