TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**PROJECT I**

NGÀNH CÔNG NGHỆ THÔNG TIN

**XÂY DỰNG ỨNG DỤNG WEB BÁN TRUYỆN ONLINE**

Sinh viên thực hiện: **Đinh Hoàng Nam**

Lớp CNTT2.03-K61

Giáo viên hướng dẫn: ThS. Nguyễn Đức Tiến

HÀ NỘI 12-2018

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN MÔN HỌC

**1. Thông tin về sinh viên**

Họ và tên sinh viên: Đinh Hoàng Nam

Điện thoại liên lạc: 0348768568 Email:nam.dh162793@sis.hust.edu.vn

Lớp: CNTT2.03 Hệ đào tạo: đại học

Thời gian làm đồ án: Từ ngày 03/10/2018 đến ngày 21/12/2018

**2. Mục đích nội dung của đồ án**

* Tìm hiểu về công nghệ Web
* Biết cách xây dựng một ứng dụng Web bán truyện tranh online

**3. Các nhiệm vụ cụ thể của ĐATN**

* Tìm hiểu HTML, CSS, JS…
* Tìm hiểu các tích hợp Mobiries4
* Tìm hiểu mô hình MVC
* Phân tích, Thiết kế Cơ sở dữ liệu
* Triển khai xây dựng hệ thống
* Kiểm thử hệ thống
* Tổng kết và đánh giá.

**4. Lời cam đoan của sinh viên:**

Tôi – *Đinh Hoàng Nam* - cam kết đồ án là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của giáo viên. Các kết quả nêu trong đồ án là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 10 tháng 12 năm 2018*  Tác giả đồ án  Đinh Hoàng Nam |

**5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của đồ án**

|  |  |
| --- | --- |
|  | *Hà Nội, ngày … tháng … năm …..*  Giáo viên hướng dẫn |

MỤC LỤC

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN MÔN HỌC 2](#_Toc514714239)

[PHẦN I: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP 8](#_Toc514714240)

[1.1. Tổng quan 8](#_Toc514714241)

[1.1.1. Các vấn đề, khó khăn hiện tại 8](#_Toc514714242)

[1.1.2. Mục tiêu cần đạt được 8](#_Toc514714243)

[1.1.3. Lựa chọn và định hướng thiết kế 8](#_Toc514714244)

[1.2. Thuật toán 8](#_Toc514714245)

[1.3. Các phần cứng sử dụng trong hệ thống 8](#_Toc514714246)

[1.4. Các phần mềm tích hợp 9](#_Toc514714247)

[1.5. Các framework/thư viện lập trình 9](#_Toc514714248)

[1.6. Giao thức 9](#_Toc514714249)

[1.7. Ứng dụng các kỹ thuật vào bài toán đã đề ra 9](#_Toc514714250)

[PHẦN II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 10](#_Toc514714251)

[2.1. Các chức năng 10](#_Toc514714252)

[2.2. Usecase 10](#_Toc514714253)

[2.3. Thuật toán cốt lõi 10](#_Toc514714254)

[2.4. Thiết kế cơ sở dữ liệu 11](#_Toc514714255)

[2.5. Các lớp đối tượng 14](#_Toc514714256)

[2.6. Thiết kế giao diện/layout 14](#_Toc514714257)

[2.7. Mô hình tích hợp phần cứng/phần mềm trong hệ thống 15](#_Toc514714258)

[PHẦN III: CÀI ĐẶT VÀ TRIỂN KHAI GIẢI PHÁP 16](#_Toc514714259)

[3.1. Tính năng 1 16](#_Toc514714260)

[3.2. Tính năng 2 16](#_Toc514714261)

[3.3. Giao diện 1 16](#_Toc514714262)

[3.4. Giao diện 2 16](#_Toc514714263)

[PHẦN IV: KẾT LUẬN 17](#_Toc514714264)

[TÀI LIỆU THAM KHẢO 18](#_Toc514714265)

DANH MỤC BẢNG

[Bảng 1: Danh mục các từ viết tắt và thuật ngữ 6](#_Toc533112233)

[Bảng 2: bảng user 12](#_Toc533112234)

[Bảng 3: bảng bill 12](#_Toc533112235)

[Bảng 4: bảng billdetail 12](#_Toc533112236)

[Bảng 5: bảng book 12](#_Toc533112237)

[Bảng 6: bảng creation 13](#_Toc533112238)

[Bảng 7: bảng category 13](#_Toc533112239)

[Bảng 8: bảng author 13](#_Toc533112240)

[Bảng 9: bảng publisher 13](#_Toc533112241)

[Bảng 10: bảng inputevent 13](#_Toc533112242)

[Bảng 11: bảng book\_inputevent 14](#_Toc533112243)

[Bảng 12: bảng saleevent 14](#_Toc533112244)

[Bảng 13; bảng book\_saleevent 14](#_Toc533112245)

DANH MỤC HÌNH VẼ

[Hình 1: Biểu đồ phân rã chức năng 9](#_Toc533111995)

[Hình 2: Biểu đồ use case 10](file:///C:\Users\Hoang%20Nam\Desktop\MyProject.docx#_Toc533111996)

[Hình 3: Sơ đồ quan hệ thực thể 11](#_Toc533111997)

[Hình 4 Các module chính trong hệ thống 15](#_Toc533111998)

[Hình 5: Biểu đồ lớp của chức năng Đăng nhập 16](#_Toc533111999)

[Hình 6: Biểu đồ lớp của chức năng quản trị nội dung 16](#_Toc533112000)

[Hình 7: Mô hình tích hợp phần mềm, phần cứng 17](#_Toc533112001)

[Hình 8: Tính năng Đăng kí 18](#_Toc533112002)

[Hình 9: Tính năng quản trị nội dung 21](#_Toc533112003)

[Hình 10: Giao diện thông tin tài khoản người dùng 24](file:///C:\Users\Hoang%20Nam\Desktop\MyProject.docx#_Toc533112004)

[Hình 11: Giao diện thông tin chi tiết về sách 24](#_Toc533112005)

# DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ

|  |  |  |
| --- | --- | --- |
| **Số thứ tự** | **Từ viết tắt** | **Ý nghĩa** |
| 1 | ĐA | Đồ án |
| 2 | OS | Operating System – Hệ điều hành |
| 3 | SDK | Software Development Kit – Bộ công cụ phát triển phần mềm |

Bảng : Danh mục các từ viết tắt và thuật ngữ

# PHẦN I: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP

## Tổng quan

### Các vấn đề, khó khăn hiện tại

1. Chưa có kiến thức cơ bản về Web
2. Thiếu kiến thức HTML, CSS, JS
3. Chưa được học về Database, SQL
4. Chưa biết ngôn ngữ Java hay PHP
5. Không biết sử dụng Framework
6. Kỹ năng lập trình kém
7. Chưa được học phân tích, thiết kế hệ thống
8. Không có kiến thức về biểu đồ lớp, biểu đồ usecase,…

### Mục tiêu cần đạt được

1. Nắm được cơ bản HTML, CSS, JS
2. Hiểu thêm Bootstrap, jQuery
3. Hiểu mô hình MVC
4. Biết cách sử dụng database
5. Biết dùng một framework
6. Xây dựng được một ứng dụng đơn giản

### Lựa chọn và định hướng thiết kế

1. Xây dựng ứng dụng web theo mô hình MVC
2. Sử dụng ngôn ngữ Java và Spring framework
3. Tích hợp Mobirise4 làm front-end
4. Dùng cơ sở dữ liệu quan hệ

## Thuật toán

1. Hệ thống không yêu cầu sử dụng đến thuật toán phức tạp.

## Các phần cứng sử dụng trong hệ thống

1. Laptop cá nhân

## Các phần mềm tích hợp

1. Java Development Kit 1.8
2. MySQL 5.7
3. Spring Tool Suite STS 3.9
4. TomCate 9
5. Mobirise 4

## Các framework/thư viện lập trình

1. Spring MVC
2. Spring Data Jpa
3. Hibernate

# PHẦN II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Các chức năng

*Biểu đồ phân rã chức năng*

**A screenshot of a cell phone

Description automatically generated**

Hình : Biểu đồ phân rã chức năng

## Usecase

*Biểu đồ Usecase*

*A picture containing text, map

Description automatically generated*

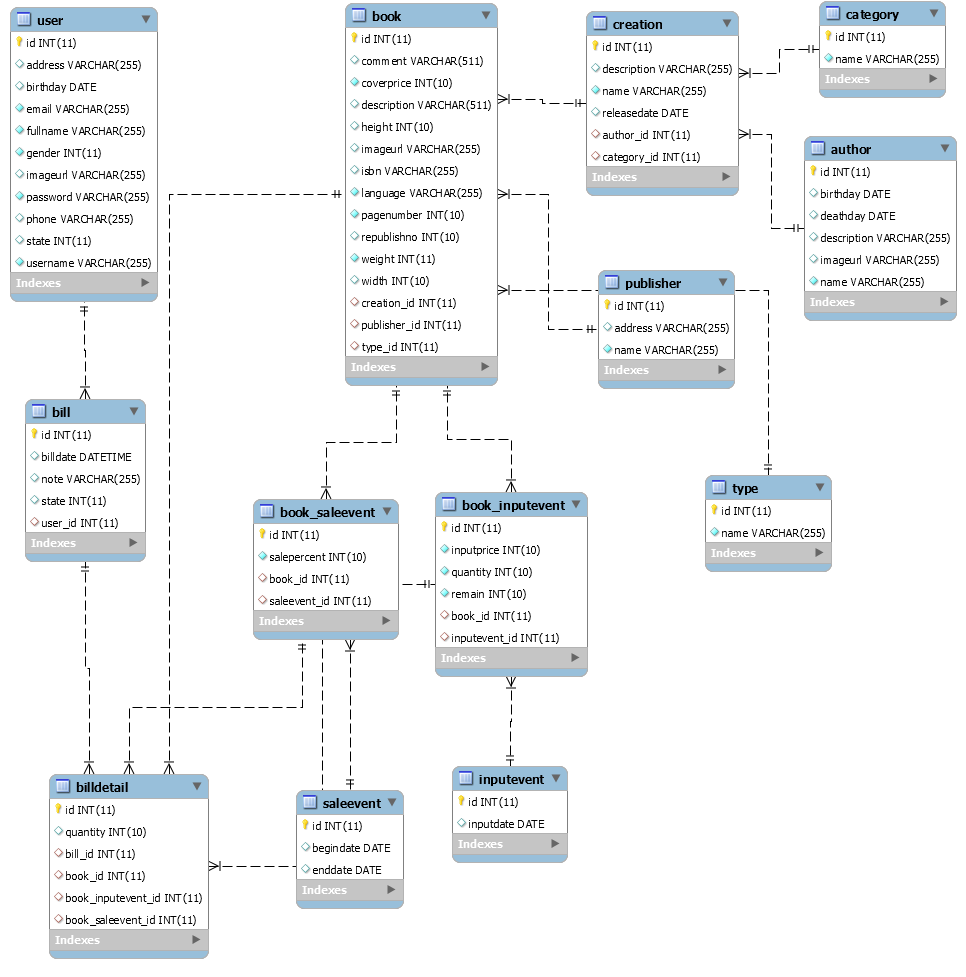
Hình : Biểu đồ use case

## Thuật toán cốt lõi

Hệ thống được thiết kế theo mô hình MVC:

Các sự kiện yêu cầu từ phía người dùng sẽ được gửi đến Controller. Ở đây Controller sẽ gọi đến dịch vụ tương ứng của các lớp Service để thực hiện logic nghiệp vụ, từ đó, các lớp Service sẽ gọi các hàm thích hợp của các lớp Repository để lấy dữ liệu trả về cho Controller và Controller sẽ gửi dữ liệu này tới các View JSP để trả kết quả về cho người dùng.

## Thiết kế cơ sở dữ liệu

*Sơ đồ quan hệ thực thể - ERD*

Hình 3: Sơ đồ quan hệ thực thể

*Đặc tả rõ các bảng dữ liệu có trong cơ sở dữ liệu*

Bảng user: Danh sách người sử dụng

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | ID người dùng |
| username | varchar(255) | Tên đăng nhập |
| password | varchar(255) | Mật khẩu |
| email | varchar(255) | Email người dùng |
| fullname | varchar(255) | Tên đầy đủ |
| gender | int(11) | Giới tính |
| birthday | date | Ngày sinh |
| address | varchar(255) | Địa chỉ |
| phone | varchar(255) | Số điện thoại |
| imageurl | varchar(255) | Link dẫn đến ảnh đại diện |
| state |  | Trạng thái (0 – active, 1 – block) |
|  |  |  |

Bảng 2: bảng user

Bảng bill: hóa đơn mua hàng

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | ID hóa đơn |
| billdate | date | Ngày đặt hóa đơn |
| note | varchar(255) | Ghi chú (thời gian giao hàng, địa chỉ dự phòng …) |
| state | int(11) | Trạng thái hóa đơn (đang giao hàng, đã giao hàng, đã hủy..) |

Bảng : bảng bill

Bảng billdetail: chi tiết mặt hàng và số lượng đặt mua

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | ID chi tiết hóa đơn |
| quantity | varchar(255) | Số lượng mặt hàng |
| email | varchar(255) | Mô tả |

Bảng : bảng billdetail

Bảng book: thông tin về sách

Bảng type: kiểu sách (bìa cứng, bìa mềm, tranh màu, light novel..)

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| name | varchar(255) | Tên đầy đủ |

Bảng : bảng book

Bảng creation: nguyên tác (một creation có thể có một số book tương ứng)

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| name | varchar(255) | Tên đầy đủ |
| releasedate | date | Mô tả |
| description | varchar(255) |  |

Bảng : bảng creation

Bảng category: Danh mục sách (Hành động, Truyện vui, Trinh thám…)

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| name | varchar(255) | Tên đầy đủ |

Bảng : bảng category

Bảng author: tác giả

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| name | varchar(255) | Tên đầy đủ |
| birthday | date | Mô tả |
| deathday | date |  |
| description | varchar(255) |  |
| imageurl | varchar(255) |  |

Bảng : bảng author

Bảng publisher: nhà xuất bản

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| name | varchar(255) | Tên đầy đủ |
| address | varchar(255) | Mô tả |

Bảng : bảng publisher

Bảng inputevent: ngày nhập sách

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| inputdate | date | Tên đầy đủ |

Bảng : bảng inputevent

Bảng book\_inputevent: thông tin về đợt nhập sách

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| inputprice | int(11) | Tên đầy đủ |
| quantity | int(11) | Số lượng sách tương ứng nhập |
| remain | int(11) | Số lượng sách tương ứng còn lại |

Bảng : bảng book\_inputevent

Bảng saleevent: thông tin về ngày khuyến mại

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| begindate | date | ngày bắt đầu khuyến mại |
| enddate | date | ngày kết thúc khuyến mại |

Bảng : bảng saleevent

Bảng Book\_SaleEvent: thông tin về ngày khuyến mại

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Ghi chú |
| id | int(11) | Id |
| salepercent | int(11) | Tên đầy đủ |

Bảng ; bảng book\_saleevent

## Các lớp đối tượng

*Các module chính trong hệ thống*

*A screenshot of a cell phone

Description automatically generated*

Hình Các module chính trong hệ thống

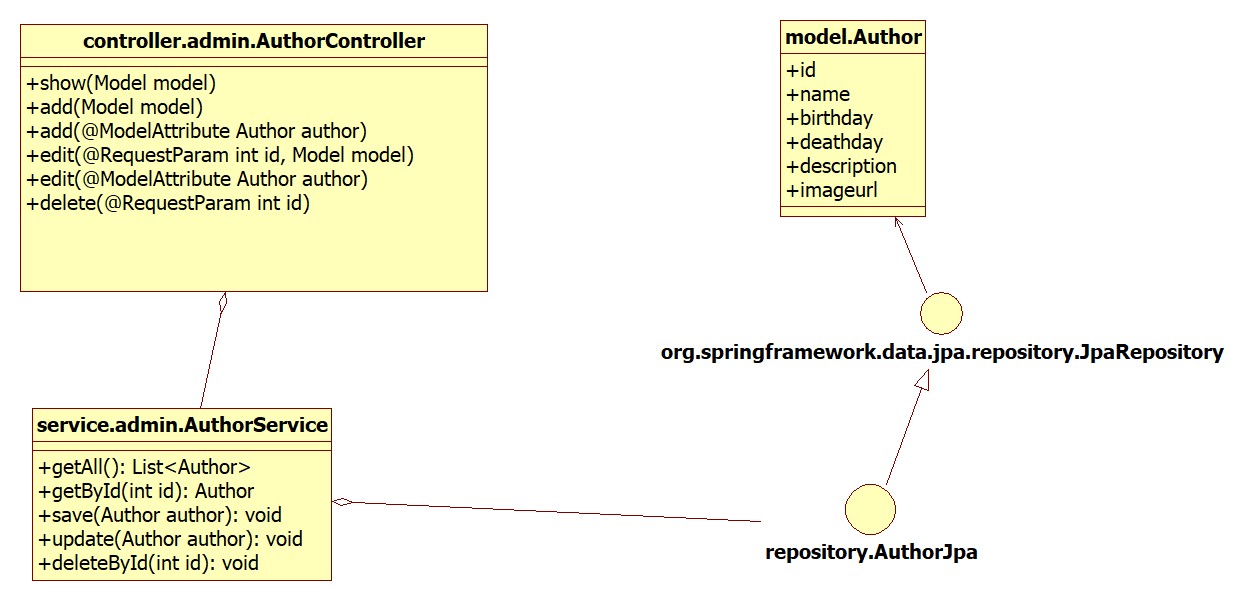
*Biểu đồ lớp thể hiện chức năng đăng kí của user*

*A screenshot of text

Description automatically generated*

Hình : Biểu đồ lớp của chức năng Đăng nhập

*Biểu đồ lớp đại diện cho chức năng quản trị nội dung của admin*

**

Hình : Biểu đồ lớp của chức năng quản trị nội dung

## Thiết kế giao diện/layout

## Mô hình tích hợp phần cứng/phần mềm trong hệ thống

Hình 7: Mô hình tích hợp phần mềm, phần cứng

# PHẦN III: CÀI ĐẶT VÀ TRIỂN KHAI GIẢI PHÁP

* Hệ thống đã hoàn thiện tính năng đăng kí, đăng nhập, xem, sửa thông tin tài khoản cho user
* Hệ thống đã hoàn thiện tính năng xem thông tin chi tiết về sách
* Hệ thống đã hoàn thiện tính năng quản trị nội dung, quản lý nhập xuất kho cho admin
* Hệ thống chưa hoàn thiện tính năng mua hàng
* Hệ thống chưa hoàn thiện tính năng bảo mật

## Tính năng đăng ký tài khoản (user)

*Giao diện tính năng đăng ký tài khoản:*

*A screenshot of a social media post

Description automatically generated*

Hình : Tính năng Đăng kí

*Luồng thực hiện tuần tự của chức năng đăng kí:*

b1: người dùng nhấp link đăng kí trên thanh header

b2: hệ thống hiển thị form đăng kí

b3: người dùng nhập thông tin và submit

b4: hệ thống kiểm tra tính hợp lệ:

* nếu đăng kí thành công, tạo và lưu trữ tài khoản, trả về trang thông tin tài khoản cho người dùng
* nếu không hợp lệ, trả về trang báo lỗi

*Đoạn mã nguồn xử lý tương ứng:*

package knh.t7.controller.user;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import knh.t7.model.User;

import knh.t7.service.user.SignUpService;

@Controller

@RequestMapping("user/")

public class SignUpController {

@Autowired

private SignUpService signUpService;

@GetMapping("signUp")

public String signUp(Model model) {

model.addAttribute("user", new User());

return "user/signUp";

}

@PostMapping("signUp")

public String signUp(@ModelAttribute("user") User user, Model model) {

int retvl = signUpService.signUp(user);

if (retvl == -1) {

return "redirect:/user/info" + "?id=" + user.getId();

}

model.addAttribute("errorCode", retvl);

return "redirect:/user/signUpFailure";

}

@GetMapping("signUpFailure")

public String showSignUpFailure(Model model) {

return "user/signUpFailure";

}

}

*Giải thích đoạn mã quan trọng:*

SignUpController bắt sự kiện người dùng nhấn link signUp về trả về một form để người dùng nhập thông tin.

@GetMapping("signUp")

public String signUp(Model model) {

model.addAttribute("user", new User());

return "user/signUp";

}

SignUpController bắt sự kiện người dùng submit form, đóng gói thông tin thành đối tượng user, gọi hàm signUp từ signUpService để kiểm tra tính hợp lệ, nếu đăng kí thành công, chuyển điều hướng sang trang thông tin người dùng, ngược lại, chuyển sang trang đăng kí thất bại.

@PostMapping("signUp")

public String signUp(@ModelAttribute("user") User user, Model model) {

int retvl = signUpService.signUp(user);

if (retvl == -1) {

return "redirect:/user/info" + "?id=" + user.getId();

}

model.addAttribute("errorCode", retvl);

return "redirect:/user/signUpFailure";

}

## Tính năng quản trị nội dung (admin)

*Giao diện tính năng quản trị nội dung:*

A screenshot of a computer screen

Description automatically generated

Hình : Tính năng quản trị nội dung

*Luồng thực hiện tuần tự của hệ thống với chức năng quản trị nội dung :*

b1: admin đăng nhập với tài khoản admin

b2: hệ thống kiểm tra đăng nhập, nếu thành công, trả về trang quản trị cho admin

b3: admin chọn nội dung cần quản trị (Author, User, Input, Sale…)

b4: hệ thống trả về bảng chứa thông tin tương ứng

b5: admin có thể tìm kiếm ở ô “Search”

b6: admin có thể nhấn “Add” để thêm, “Edit” để sửa, “Delete” để xóa.

*Đoạn mã nguồn tương ứng:*

package knh.t7.controller.admin;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestParam;

import knh.t7.model.Author;

import knh.t7.service.admin.AuthorService;

@Controller

@RequestMapping("admin/author/")

public class AuthorController {

@Autowired

private AuthorService authorService;

@GetMapping(value= {"", "show"})

public String show(Model model) {

List<Author> listAuthor = authorService.getAll();

model.addAttribute("listAuthor", listAuthor);

return "admin/author/show";

}

@GetMapping("add")

public String add(Model model) {

model.addAttribute("author", new Author());

return "admin/author/add";

}

@PostMapping("add")

public String add(@ModelAttribute("author") Author author) {

authorService.save(author);

// here need replace

return "redirect:/admin/author/add";

}

@GetMapping("edit")

public String edit(@RequestParam("id") int id, Model model) {

Author author = authorService.getById(id);

model.addAttribute("author", author);

return "admin/author/edit";

}

@PostMapping("edit")

public String editAuthor(@ModelAttribute("author") Author author) {

authorService.update(author);

return "redirect:/admin/author/";

}

@GetMapping("delete")

public String delete(@RequestParam("id") int id) {

authorService.deleteById(id);

return "redirect:/admin/author/";

}

}

*Giải thích một số đoạn mã quan trọng:*

Controller nhận danh các các Author từ Service và trả về cho Model khi admin nhấn link tương ứng:

@GetMapping(value= {"", "show"})

public String show(Model model) {

List<Author> listAuthor = authorService.getAll();

model.addAttribute("listAuthor", listAuthor);

return "admin/author/show";

}

Controller nhận yêu cầu sửa thông tin về Author với id tương ứng và trả về một form với các thông tin của Author đó:

@GetMapping("edit")

public String edit(@RequestParam("id") int id, Model model) {

Author author = authorService.getById(id);

model.addAttribute("author", author);

return "admin/author/edit";

}

Controller nhận yêu cầu xóa Author từ admin, và gọi đến hàm deleteById từ authorService để thực thi yêu cầu:

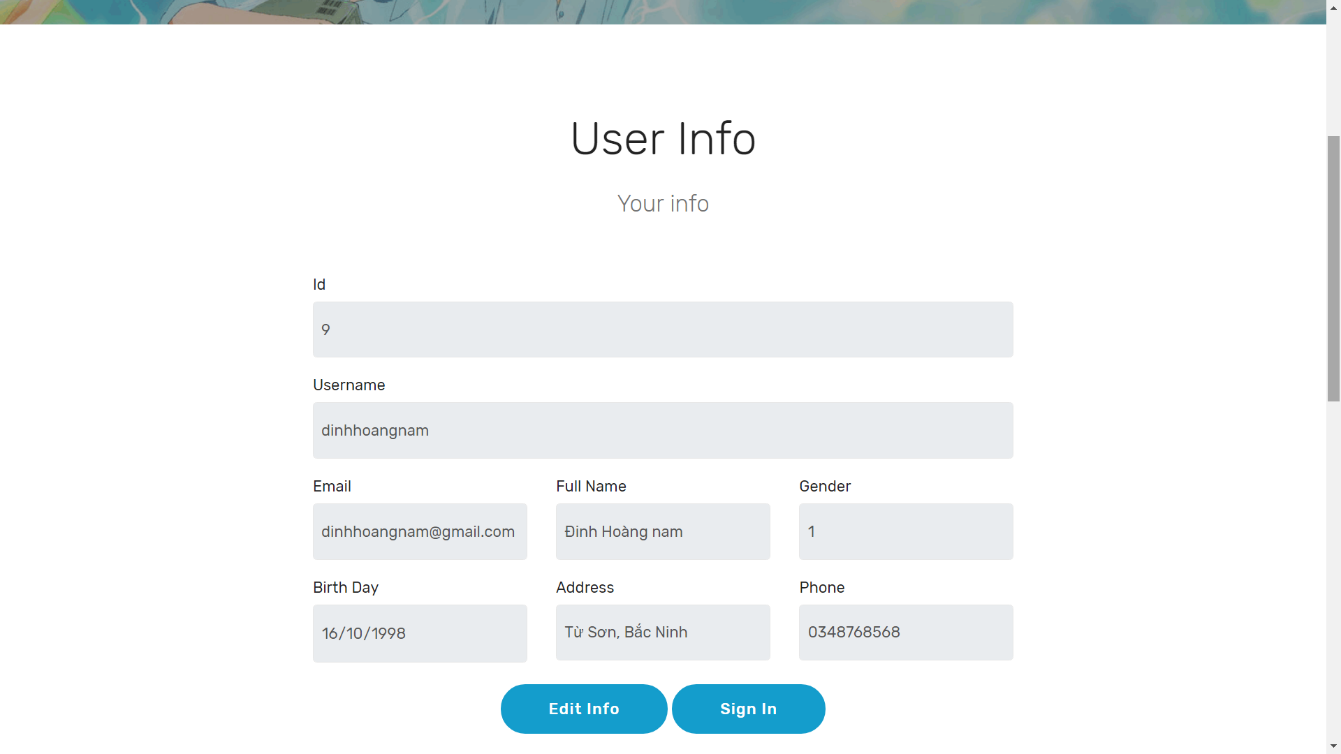
@GetMapping("delete")

public String delete(@RequestParam("id") int id) {

authorService.deleteById(id);

return "redirect:/admin/author/";

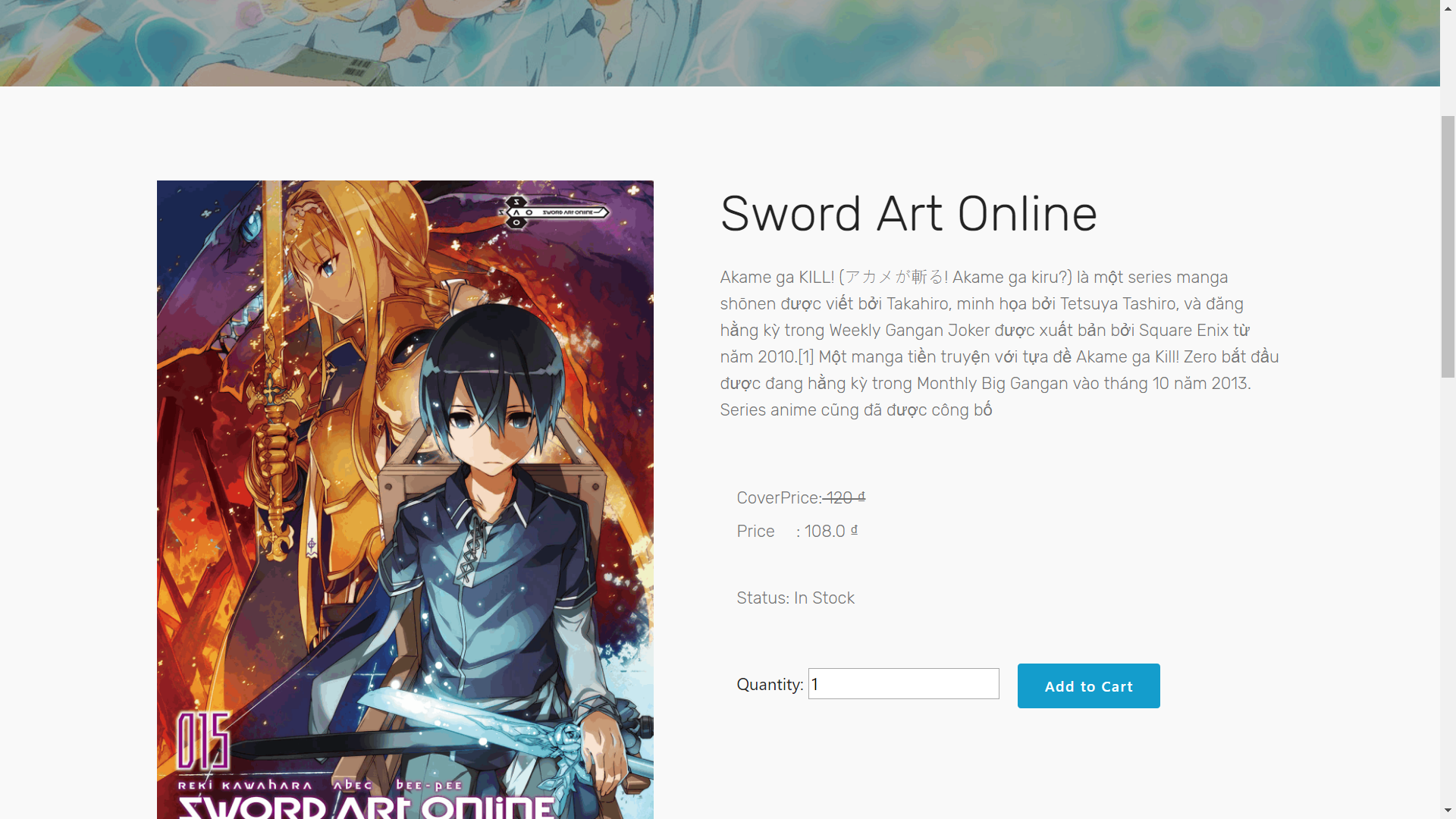
}

*Giao diện xem sửa thông tin user*

Hình : Giao diện thông tin tài khoản người dùng

## Giao diện thông tin về truyện

*Giao diện xem thông tin chi tiết về truyện*



Hình : Giao diện thông tin chi tiết về sách

# PHẦN IV: KẾT LUẬN

*Quá trình thực hiện đồ án đã gặp phải rất nhiều khó khăn:*

* Tiếng anh còn hạn chế nên đọc tài liệu gặp nhiều trở ngại.
* Thiếu kiến thức cơ bản về web nên phải tìm hiểu rất nhiều khái niệm cơ bản
* Học về front-end phải nhớ rất nhiều thứ, và dễ quên nếu lâu không dùng
* Phải liên tục bổ sung kiến thức của các môn cơ bản khác mà quá trình thực hiện đề án cần tới như: Lập trình hướng đối tượng, Cơ sở dữ liệu, Mạng máy tính…
* Khó khăn khi bắt đầu học các framework
* Quá trình phát triển hệ thống gặp rất rất nhiều lỗi của việc sử dụng các công cụ, đều là các lỗi nhỏ, nhưng rất khó hiểu, và đều mất rất nhiều thời gian để tìm hiểu và giải quyết
* Nhiều lần cảm thấy học tập rất áp lực
* Mặc dù năng lực còn hạn chế, chưa làm được nhiều trong đồ án, nhưng đã có rất nhiều nỗ lực trong quá trình học tập, và đã học được rất nhiều điều, và cảm thấy vui với những gì mình đã đạt được.

*- Định hướng nhằm phát triển, hoàn thiện hệ thống:*

* Dùng SpringSecurity để đảm bảo tính an toàn, bảo mật cho hệ thống.
* Mã hóa mật khẩu để đảm bảo an toàn.
* Bổ sung chức năng tìm kiếm sách theo tên.
* Phát triển giao diện người dùng: thêm một card chứa các danh mục “Tác giả”, “Nhà xuất bản”, “Thể loại” … giúp thuận tiện hơn cho người dùng.

# TÀI LIỆU THAM KHẢO

**\* *Danh mục sách:***

1. Gall, J-C. Paléoécologie. Paysages et environnements disparus. 2e éd. Paris: Masson, 1998. 239p. ISBN 2- 225-83084-3

**\* *Danh mục tạp chí:***

1. Deleu, M. *et al*. Apercu des techniques d’analyse conformationelle des macromolecules biologiques. Biotechnologie, Agronomie, Societé et Environnement, 1998, vol 2, no 4, p.234-247

**\* *Danh mục hội thảo:***

1. Troxler, W.L. Thermal desorption. In Kearney, P. and Roberts, T. (eds), Pesticide remediation in soils and water. Chichester, UK: Wiley, 1998, p.105-128

**\* *Danh mục internet:***

1. Qui định về đồ án tốt nghiệp của Viện CNTT-TT  
   <https://soict.hust.edu.vn/index.php/2017/03/16/quy-dinh-ve-do-an-tot-nghiep/>
2. CodeProject, [www.codeproject.com](http://www.codeproject.com).
3. Ashby J.A et al. Investing in Farmers as Researchers. Ciat publication n0 318 [online]. Cali, Colombia: CIAT, 2000 [ref. on Jan 20th 2002). <http://www.ciat.cigiar.org/downloads/pdf/Investing_farmers.pdf>