

**ĐỒ ÁN JAVA**

**XÂY DỰNG WEB QUẢN LÝ KHÁCH SẠN**

**(JAVA SPRING MVC JFS)**

**Ngành: CÔNG NGHỆ THÔNG TIN**

**Giảng viên hướng dẫn : ThS. VÕ TẤN DŨNG**

**Sinh viên thực hiện :**

**BÙI VĂN BẢO**

**MSSV: 1811062384**

**Lớp: 18DTHC1**

**ĐẶNG VŨ PHƯƠNG NAM**

**MSSV: 1811061659**

**Lớp: 18DTHC1**

**TP. Hồ Chí Minh, 2020**

## LỜI MỞ ĐẦU

Những website Khách sạn giúp mọi người có thể đặt phòng cũng như tìm được phòng với giá cả phù hợp với bản thân, ở mọi lúc mọi nơi có Internet; giúp người dùng gửi và nhận phòng nhanh chóng một cách trực tiếp với khách sạn. giúp mọi người dùng có thể yên tâm đến du lịch không lo đến việc không có phòng. trong việc tìm kiếm và đặt chỗ trong một khách sạn mà có thể đáp ứng được nhu cầu của mình, trong khi đó thì hệ thống các khách sạn ở nước ta ngày càng lớn, hiện đại và tiện nghi.

Với mong muốn có thể tạo được một hệ thống website cho phép người dùng tìm kiếm thông tin về các khách sạn ở Việt Nam, đồng thời hỗ trợ việc đặt phòng trước qua mạng, chúng em đã tập trung tìm hiểu và thực hiện đề tài “XÂY DỰNG WED ĐẶT PHÒNG KHÁCH SẠN”. Mặc dù đây không hẳn là một nội dung mới mẻ (vì đã có rất nhiều trang web hỗ trợ các dịch vụ này như: <https://www.booking.com>, <http://www.saigon-hotels.net...>).

Đặt biệt em xin gửi lời cảm ơn đến thầy Võ Tấn Dũng, thầy đã hỗ trợ tận tình trong suốt quá trình làm đồ án môn học. Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một học viên, bài báo cáo này không thể tránh được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các quý thầy cô để tôi có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn công tác thực tế sau này.

Nhóm chúng em xin chân thành cảm ơn!

Sinh viên thực hiện:

**BÙI VĂN BẢO**

**ĐẶNG VŨ PHƯƠNG NAM**

# MỤC LỤC

LỜI MỞ ĐẦU.....	2
CHƯƠNG 1. TỔNG QUAN.....	6
1.1. Tổng quan về vấn đề được nghiên cứu.....	6
1.2. Nhiệm vụ đồ án.....	6
1.2.1. Lý do hình thành đồ án.....	6
1.2.2. Mục tiêu nghiên cứu.....	6
1.2.3. Đối tượng.....	6
1.2.4. Phạm vi giới hạn.....	6
1.3. Cấu trúc đồ án.....	7
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT.....	7
2.1. Các khái niệm và cơ chế hoạt động.....	7
2.1.1. Công nghệ Java: Spring MVC JSF (Spring Web MVC).....	9
2.1.2. JSP (JavaServer Pages).....	10
2.1.3. JDBC (Java Database Connectivity).....	12
2.1.4. Microsoft SQL Server.....	15
CHƯƠNG 3. PHÂN TÍCH.....	16
3.1. Thiết kế dữ liệu.....	16
3.1.1 Mô hình dữ liệu dạng logic.....	16
3.1.2 Danh sách các bảng dữ liệu.....	16
3.1.3 Danh sách thuộc tính của các bảng.....	17
CHƯƠNG 4. KẾT QUẢ THỰC NGHIỆM.....	20
4.1. Các thành phần chức năng của trang web.....	20
4.2 Giao diện trực quan.....	20
4.2.1.Giao diện quản lý.....	20
4.2.2. Giao diện khách truy cập.....	23

CHƯƠNG 5. KẾT LUẬN KIẾN NGHỊ.....	25
4.1. Kết luận chung.....	25
4.2. Hướng phát triển và mở rộng đề án.....	25
TÀI LIỆU THAM KHẢO.....	26

## Danh sách bảng

3-1 Bảng danh sách các bảng dữ liệu.....	16
3-2. Bảng TaiKhoan.....	17
3-3. Bảng kháchSan.....	17
3-4. Bảng LoaiKhachSan.....	18
3-5. Bảng DatPhong.....	18
3-6. Bảng Phong.....	19
3-7. Bảng ThanhPho.....	19

## Danh sách hình

Hình 2.1. Mô hình Spring MVC.....	8
Hình 2.2. Flow trong Spring MVC.....	9
Hình 2.3. Vị trí JSP trong một ứng dụng web.....	11
Hình 2-1. Diagram Web .....	16
Hình 3-1 giao diện admin quản lý tài khoản.....	20
Hình 3-2 giao diện admin quản lý thành phố.....	20
Hình 3-3 giao diện admin quản lý loại khách sạn.....	20
Hình 3-4 giao diện admin quản lý khách sạn.....	21
Hình 3-5 giao diện admin quản lý phòng.....	21
Hình 3-6 giao diện khách truy cập trang chủ.....	22
Hình 3-7 giao diện khách truy cập khách sạn để đặt phòng.....	23
Hình 3-8 giao diện khách truy cập thông tin cá nhân.....	23
Hình 3-9 giao diện khách truy cập thông tin cá nhân.....	23
Hình 3-10 giao diện khách truy cập lịch sử đặt phòng.....	22

# **CHƯƠNG 1. TỔNG QUAN**

## **1.1. Tổng quan về vấn đề được nghiên cứu**

Hiện nay, nhu cầu thông tin liên lạc phát triển, nhu cầu trao đổi thông tin nhanh và hiệu quả ngày càng gia tăng. Cùng với đó là sự phát triển của công nghệ thông tin và hệ thống mạng Internet. Phải làm thế nào để mọi người có thể tìm được các khách sạn và các phòng có giá hợp lý phù hợp với bản thân. Một cách nhanh chóng và tiện lợi? Với các yếu tố trên, yêu cầu về một loại hình trao đổi thông tin nhanh đơn giản hiệu quả phù hợp về mặt sử dụng, được viết trên nền web được đặt ra. Bài toán trên đặt ra là làm sao có thể xây dựng một website cho phép khách hàng có thể tìm kiếm thông tin của các khách sạn trong nước với các nội dung cụ thể như: Tên khách sạn, địa chỉ, chuẩn khách sạn, số phòng, giá từng loại phòng... Ngoài ra, nếu tìm được khách sạn mong muốn họ có thể đặt phòng trực tiếp qua mạng .

Trang web phải giúp mọi người có thể đặt phòng cũng như tìm được phòng với giá cả phù hợp với bản thân, ở mọi lúc mọi nơi có Internet.

## **1.2. Nhiệm vụ đề án**

### **1.2.1. Lý do hình thành đề án**

Ngày nay, chúng ta đang chứng kiến sự phát triển của công nghiệp thông tin nói chung và Internet nói riêng, cùng với sự phát triển đó cộng với nhu cầu trao đổi và thông tin liên lạc của mọi người một cách nhanh chóng và tiện lợi đã thúc đẩy sự phát triển các trang web để trao đổi những thông tin một cách trực tiếp và tức thì.

Từ những vấn đề trên, chúng ta xây dựng một trang web phục vụ người dùng trao đổi trực tiếp với các tổ chức cá nhân một cách tức thì và nhanh chóng.

### **1.2.2. Mục tiêu nghiên cứu**

Mục tiêu của đề án này là giúp chúng em học được một số ngôn ngữ mới, một số công nghệ mới, giúp chúng em củng cố kiến thức lập trình. Những kiến thức đó sẽ giúp ích chúng em sau này và có thể áp dụng được vào những đề án tiếp theo.

### 1.3. Cấu trúc đồ án

#### Chương 1: Tổng quan

- *Tổng quan về vấn đề được nghiên cứu:* Tóm tắt những lý thuyết, tài liệu có liên quan đến đề tài “*Xây dựng web quản lý khách sạn*”.
- *Nhiệm vụ đồ án:* Lý do hình thành đề tài, mục tiêu nghiên cứu.
- *Cấu trúc đồ án:* Trình bày cấu trúc của đồ án gồm các chương và tóm tắt từng chương.

#### Chương 2: Cơ sở lý thuyết

Các khái niệm và phương pháp bao gồm các công nghệ, hệ thống, các ràng buộc,... để giải quyết nhiệm vụ của đồ án.

#### Chương 3: Thiết kế

Bao gồm các tài liệu liên quan đến thiết kế hệ thống như : thiết kế dữ liệu, thiết kế giao diện.

#### Chương 4: Kết quả thực nghiệm

Chạy thử web quản lý khách sạn trên eclicp

#### Chương 5: Kết luận và kiến nghị

Những kết luận chung, tóm tắt những kết quả đạt được và đề xuất các hướng phát triển đề tài.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. Các khái niệm và cơ chế hoạt động

#### 2.1.1. Công nghệ Java: Spring MVC (Spring Web MVC)

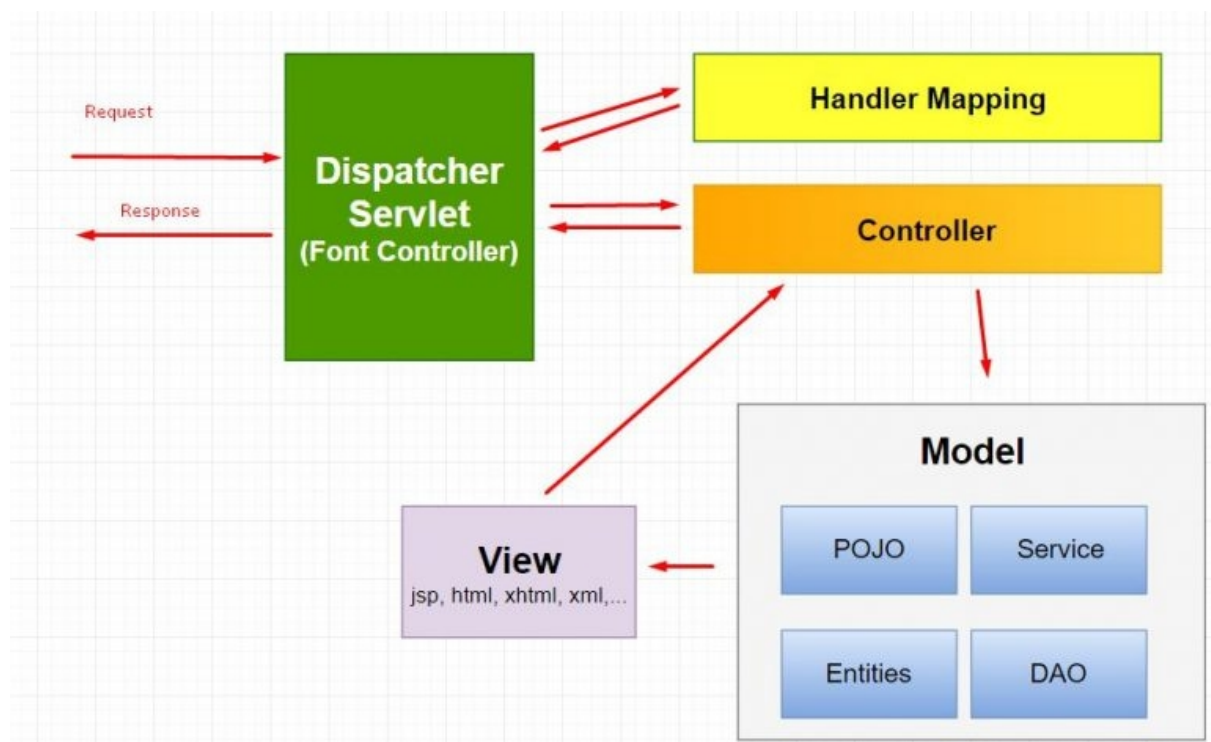
##### 2.1.1.1. Khái niệm của Spring MVC

Spring framework là nền tảng mã nguồn mở. Nó cung cấp cơ sở hạ tầng toàn diện để phát triển ứng dụng Java một cách mạnh mẽ, rất dễ dàng và nhanh chóng. Spring framework được tạo bởi Rod Johnson và được giới thiệu vào tháng 6 năm 2003. Spring là framework phát triển ứng dụng Java phổ biến nhất đối với doanh nghiệp,

được hàng triệu nhà phát triển ứng dụng trên toàn thế giới sử dụng để tạo ra các sản phẩm phần mềm với hiệu suất cao, dễ dàng kiểm chứng, tái sử dụng mã.

Spring MVC là một Framework mã nguồn mở của Spring, có các API cho phép việc xây dựng ứng dụng web được dễ dàng hơn và chuẩn hơn. Chuẩn hơn thể hiện ở chỗ mọi thành phần được tạo ra, cài đặt và vận hành tuân theo một chuẩn thiết kế thống nhất.

MVC lần lượt là ba chữ cái đầu tiên của ba từ Model, View và Controller. MVC là một mô hình ứng dụng mà ở đó các thành phần được phân tách ra thành các lớp riêng biệt với các nhiệm vụ đặc trưng.



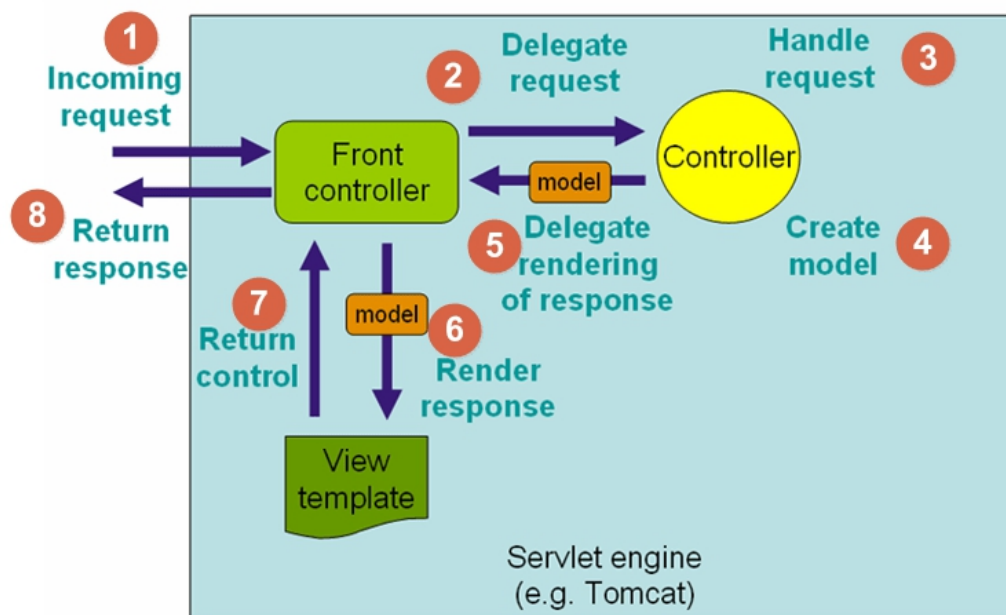
Hình 2.1. Mô hình Spring MVC

- View là lớp cho các thành phần có chức năng hiển thị, giao tiếp trực tiếp với người dùng. Nhiệm vụ của các thành phần trong View là trình bày các dữ liệu từ Model đến người dùng cuối.
- Model là các thành phần có khả năng lưu trữ và vận chuyển thông tin. Quá trình gửi dữ liệu vào Model sẽ được thực hiện bởi Controller.
- Controller là các thành phần giúp cho việc xử lý logic các thao tác nghiệp vụ. Nhiệm vụ của Controller là lấy dữ liệu từ Model, xử lý dữ liệu, và cập nhật lại dữ liệu vào Model.



Ta cụ thể hóa ý tưởng trên vào trong trường hợp của Spring MVC. Ở đây View sẽ là các trang html hay jsp giúp cho việc hiển thị dữ liệu lên trình duyệt. Dữ liệu được lấy ra từ Model là các POJO (Plain Old Java Object) hay ta còn gọi là các Domain Object. Cụ thể hơn, nó là các đối tượng có các thuộc tính có khả năng truyền dữ liệu trong các luồng nghiệp vụ của ứng dụng. Cuối cùng là Controller, là các lớp đặc biệt, có chứa các phương thức có khả năng nhận yêu cầu, xử lý yêu cầu, cập nhật dữ liệu, và chuyển tiếp dữ liệu trong ứng dụng.

#### 2.1.1.2. Flow trong Spring MVC:



Hình 2.2. Flow trong Spring MVC

- Bất kỳ request nào tới ứng dụng web đều sẽ được gửi tới Front Controller (Dispatcher Servlet)
- Front Controller sẽ sử dụng Handler Mapping để biết được controller nào sẽ xử lý request đó
- Controller nhận request, gọi tới các class service thích hợp để xử lý yêu cầu.
- Sau khi xử lý xong, Controller sẽ nhận được model từ tầng Service hoặc tầng DAO.

- Controller gửi model vừa nhận được tới Front Controller (Dispatcher Servlet)
- Dispatcher Servlet sẽ tìm các mẫu view, sử dụng view resolver và truyền model vào nó.
- View template, model, view page được build và gửi trả lại Front Controller
- Front Controller gửi một page view tới trình duyệt để hiển thị nó cho người dùng.

#### 2.1.1.3. Các lợi ích của Spring MVC

- Các tầng trong Spring MVC độc lập nên việc unit test dễ dàng hơn.
- Phần view có thể tích hợp với nhiều Framework về UI như JSF, Freemarker, Themeleaf,...
- Spring MVC dựa trên các POJO class nên các hành động của nó khá đơn giản
- Hỗ trợ cả Annotation và XML config giúp việc phát triển nhanh hơn và sạch hơn.
- Cung cấp việc phân chia một cách rõ ràng, linh hoạt giữa controller, service, data acces layer.

#### 2.1.2. JSP (JavaServer Pages)

##### 2.1.2.1. Khái niệm

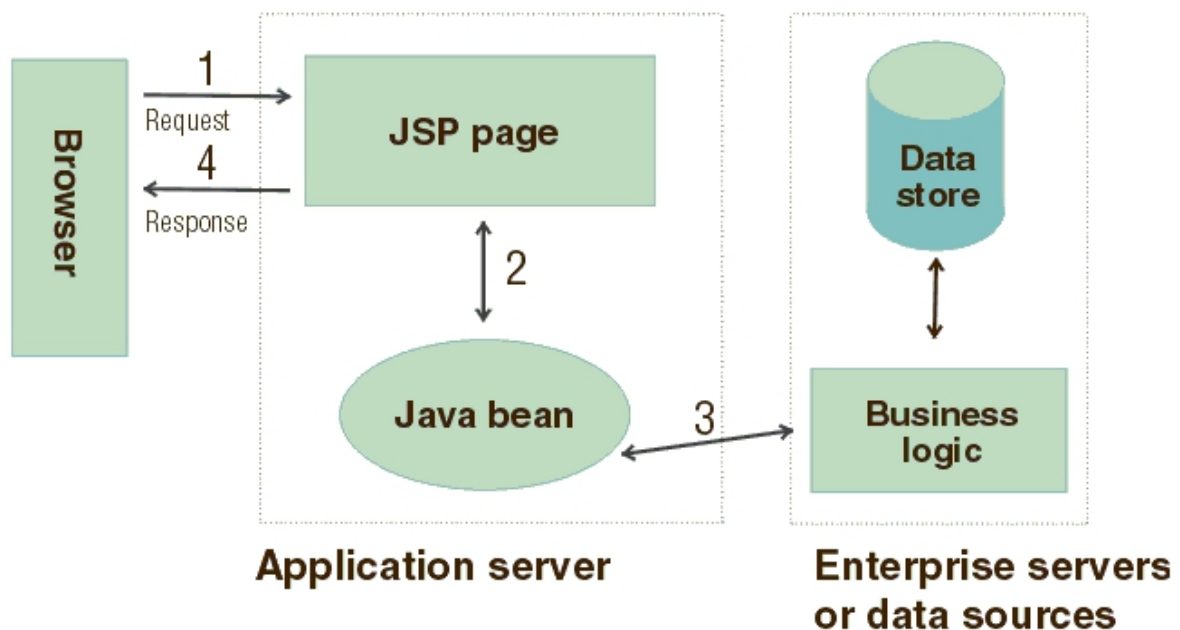
JSP là viết tắt của JavaServer Pages là một công nghệ để phát triển các trang web động. Đây là một tính năng được phát triển bởi Sun Microsystems để cải tiến cho Servlets, che đậy tất cả các lỗ hổng của Servlets. JSP giúp các nhà phát triển nhúng java code vào các trang HTML bằng cách sử dụng các thẻ JSP đặc biệt.

JSP là một kiểu Java servlet được thiết kế để tạo ra giao diện người dùng cho một ứng dụng Java web. Các nhà phát triển web viết các JSP như các tệp văn bản kết hợp mã HTML hoặc XHTML, các phần tử XML, các action và lệnh JSP.

Sử dụng JSP, bạn có thể thu thập dữ liệu đầu vào từ người dùng thông qua các Form của trang web, trình bày các bản ghi từ một cơ sở dữ liệu hoặc một nguồn khác, và tạo các trang web động.

Các thẻ JSP có thể được sử dụng cho nhiều mục đích khác nhau, chẳng hạn như truy xuất thông tin từ cơ sở dữ liệu hoặc đăng ký mới, truy cập các thành phần JavaBeans, kiểm soát giữa các trang và chia sẻ thông tin giữa các request, các trang,...

Vị trí của JSP trong một ứng dụng web:



Hình 2.3. Vị trí JSP trong một ứng dụng web

#### 2.1.2.2. Đánh giá

##### **Ưu điểm:**

- Với một phần mở rộng cho Java servlet, JSP có thể sử dụng mọi tính năng của Java Servlet. Ngoài ra, các thẻ tùy chỉnh có thể được sử dụng cùng với nó.
- Không cần phải biên dịch lại JSP khi thay đổi. Các thay đổi tự động xuất hiện khi chạy.
- Các thẻ được sử dụng rất dễ hiểu và viết.

- Hỗ trợ Java API, giờ đây bạn có thể dễ dàng sử dụng và tích hợp với mã HTML.
- Các kết quả thu được có định dạng HTML, do đó có thể được mở trên mọi trình duyệt.
- Các thẻ JSP tùy chỉnh có thể được sử dụng. Ví dụ: Thẻ có XML.
- Thay đổi có thể được thêm vào business logic page thay vì thay đổi trong mỗi trang

**Nhược điểm:**

- Mặc dù cơ sở dữ liệu có thể được truy cập bằng JSP, nhưng không dễ để truy cập cơ sở dữ liệu vì hầu hết các servlet không cung cấp hỗ trợ.
- Bản chất JSP là một servlet, nếu có một vấn đề trong code, nó rất khó để theo dõi.
- Thời gian cần thiết để biên dịch JSP lâu hơn.

### 2.1.3. JDBC (Java Database Connectivity)

JDBC là viết tắt của Java Database Connectivity là một API dùng để kết nối và thực thi các câu lệnh SQL xuống database. JDBC API sử dụng JDBC driver để làm việc với database gồm 4 loại.

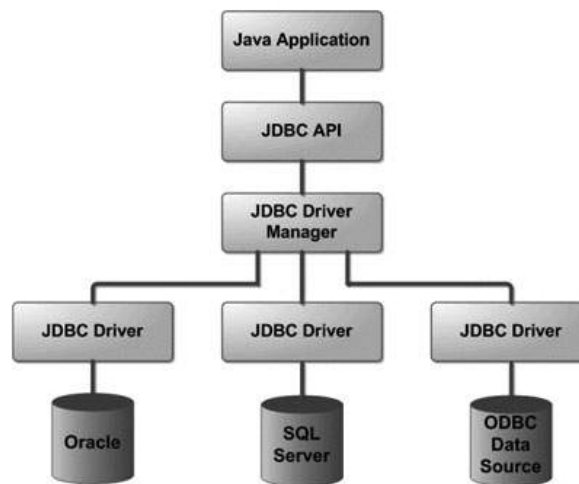
- JDBC-ODBC Bridge Driver
- Native Driver
- Network Protocol Driver
- Thin Driver

Kiến trúc của JDBC gồm 2 tầng:

- JDBC API – cho phép ứng dụng kết nối đến JDBC Manager connection.
- JDBC Driver API: Hỗ trợ JDBC Manager đến Driver connection.

JDBC API cung cấp cơ chế kết nối đến các loại database khác nhau theo một chuẩn đồng nhất.

JDBC driver đảm bảo cho JDBC API kết nối đến database mà nó cần. Đối với mỗi database sẽ có JDBC driver riêng mà JDBC API có thể chỉ định để làm việc với nó.



Hình 2.4. Kiến trúc của JDBC

Các thành phần chính trong JDBC

JDBC API cung cấp một số class và interface:

- **DriverManager** – class này chịu trách nhiệm quản lý danh sách các database driver, và chịu trách nhiệm tạo kết nối tương ứng đến database cụ thể được chỉ định.
- **Driver** – Interface chịu trách nhiệm xử lý các hoạt động giao tiếp giữa ứng dụng với database.
- **Connection** – Interface cung cấp tất cả các hàm cho việc thao tác với database.
- **Statement** – dùng các object được tạo ra từ interface này để thực thi các câu lệnh SQL xuống database.
- **ResultSet**: Lưu trữ kết quả trả về từ database.

- **SQLException** – Xử lý các lỗi xảy ra trong database.

#### 2.1.4. Microsoft SQL Server

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server...

Hệ quản trị cơ sở dữ liệu Microsoft SQL Server (MSSQL) là một trong những hệ quản trị cơ sở dữ liệu thông dụng nhất hiện nay. Đây là hệ quản trị cơ sở dữ liệu thường được sử dụng trong các hệ thống trung bình, với ưu điểm có các công cụ quản lý mạnh mẽ giúp cho việc quản lý và bảo trì hệ thống dễ dàng, hỗ trợ nhiều phương pháp lưu trữ, phân vùng và đánh chỉ mục phục vụ cho việc tối ưu hóa hiệu năng.

SQL Server luôn được Microsoft cải tiến để nâng cao hiệu năng, tính sẵn sàng của hệ thống, khả năng mở rộng và bảo mật, cung cấp nhiều công cụ cho người phát triển ứng dụng được tích hợp với bộ Visual Studio do Microsoft cung cấp. SQL Server có 4 dịch vụ lớn là Database Engine, Intergration Service, Reporting service và Analysis Services.

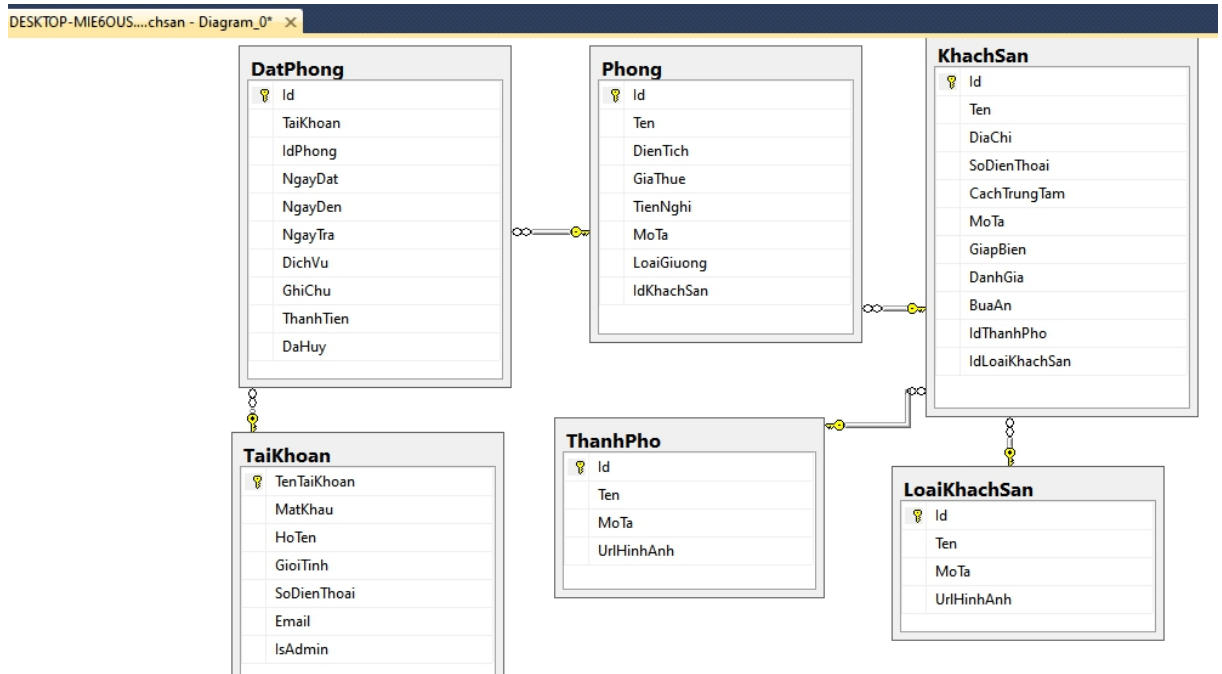
- DataBase Engine: được phát triển để thực thi tốt hơn với việc hỗ trợ cả dữ liệu có cấu trúc và dữ liệu phi cấu trúc (XML).
- Khả năng sẵn sàng của hệ thống được nâng cao, hỗ trợ các chức năng: Database mirroring (cơ sở dữ liệu gương), failover clustering, snapshots và khôi phục dữ liệu nhanh.
- Việc quản lý chỉ mục được thực hiện song song với việc hoạt động của hệ thống. Người dùng có thể thêm chỉ mục, xây dựng lại chỉ mục hay xóa một chỉ mục đi trong khi hệ thống vẫn được sử dụng.

- Chức năng phân vùng dữ liệu được hỗ trợ: Người dùng có thể phân vùng các bảng và chỉ mục cũng như quản lý phân vùng dữ liệu một cách dễ dàng. Việc hỗ trợ phân vùng dữ liệu giúp nâng cao hiệu năng hoạt động của hệ thống.
- Dịch vụ đồng bộ hóa dữ liệu được mở rộng với việc hỗ trợ mô hình đồng bộ hóa ngang hàng. Đây là dịch vụ giúp đồng bộ hóa dữ liệu giữa các máy chủ dữ liệu, giúp mở rộng khả năng của hệ thống.
- Dịch vụ tích hợp (Integration Service) thiết kế lại cho phép người dùng tích hợp dữ liệu và phân tích dữ liệu từ nhiều nguồn khác nhau. Hỗ trợ việc quản lý chất lượng dữ liệu và làm sạch dữ liệu, một công việc quan trọng trong tiến trình ETL.
- Dịch vụ phân tích dữ liệu (Analysis Service): cung cấp khung nhìn tích hợp và thống nhất về dữ liệu cho người dùng, hỗ trợ việc phân tích dữ liệu.
- Công cụ khai phá dữ liệu (Data mining) được tích hợp hỗ trợ nhiều thuật toán khai phá dữ liệu, hỗ trợ cho việc phân tích, khai phá dữ liệu và xây dựng các hệ thống hỗ trợ ra quyết định cho người quản lý.
- Dịch vụ xây dựng quản lý báo cáo (Reporting Service) dựa trên nền tảng quản trị doanh nghiệp thông minh và được quản lý qua dịch vụ web. Báo cáo có thể được xây dựng dễ dàng với ngôn ngữ truy vấn MDX. Thông qua các công cụ trên Business Intelligent, người dùng dễ dàng truy cập báo cáo và trích xuất ra nhiều định dạng khác nhau thông qua trình duyệt web.

## CHƯƠNG 3. PHÂN TÍCH

### 3.1. Thiết kế dữ liệu

#### 3.1.1 Mô hình dữ liệu dạng logic



Hình 2-1. Diagram Web

#### 3.1.2 Danh sách các bảng dữ liệu

STT	Tên	Loại	Ý nghĩa	Ghi chú
1	Dbó.TaiKhoan	Đối tượng	Thông tin về tài khoản	Cung cấp quản lý đặt phòng và quản lý
2	Dbó.KhachSan	Đối tượng	Thông tin về khách sạn	Tên địa chỉ và các dịch vụ khách sạn.
3	Dbó.LoaiKhachSan	Đối tượng	Thông tin về loại khách sạn	Các loại khách sạn.
4	Dbó.DatPhong	Đối tượng	Thông tin đặt phòng	
5	Dbó.Phong	Đối tượng	Thông tin phòng của các khách sạn	Bao gồm giá, số lượng phòng,..
6	Dbó.ThanhPho	Đối tượng	Thông tin của các thành phố	

3-1 Bảng danh sách các bảng dữ liệu



### 3.1.3 Danh sách thuộc tính của các bảng

#### 3-2. Bảng TaiKhoan.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>TenTaiKhoan</u>	varchar(50)	Khóa chính
2	MatKhau	varchar(20)	Mật khẩu
3	HoTen	nvarchar(50)	Họ tên
4	GioiTinh	bit	Nam/nữ
5	SoDienThoai	varchar(15)	Số điện thoại
6	Email	varchar(50)	Email
7	<u>IsAdmin</u>	bit	0/1

#### 3-3. Bảng kháchSan.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>Id</u>	int	Id khách sạn
2	Ten	nvarchar(100)	Tên khách sạn
3	DiaChi	nvarchar(100)	Địa chỉ
4	SoDienThoai	varchar(10)	Số điện thoại
5	CachTrungTam	int	Khoảng cách
6	MoTa	nvarchar(1000)	Thông tin
7	GiapBien	bit	
8	DanhGia	int	Đánh giá
9	BuaAn	int	Bữa ăn
10	<u>IdThanhPho</u>	int	Khóa ngoại
11	<u>IdLoaiKhachSan</u>	int	Khóa ngoại

### 3-4. Bảng LoạiKhachSan.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>Id</u>	int	Khóa chính
2	Ten	nvarchar(100)	Tên loại khách sạn
3	MoTa	nvarchar(200)	Mô tả loại khách sạn
4	UrlHinhAnh	varchar(200)	Hình loại khách sạn
5	<u>Id</u>	int	Khóa ngoại

### 3-5. Bảng DatPhong.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>Id</u>	int	Khóa chính
2	TaiKhoan	varchar(50)	Tài khoản
3	<u>IdPhong</u>	int	Số phòng
4	NgayDat	date	Ngày đặt phòng
5	NgayDen	date	Ngày đến
6	NgayTra	date	Ngày trả
7	DichVu	nvarchar(200)	Dịch vụ sử dụng
8	GhiChu	nvarchar(200)	Ghi chú
9	ThanhTien	int	Giá tiền
10	DaHuy	bit	Đã hủy

### 3-6. Bảng Phong.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>Id</u>	int	Khóa chính
2	Ten	nvarchar(100)	Tên phòng
3	DienTich	int	Diện tích
4	GiaThue	int	Giá thuê phòng
5	TienNghì	nvarchar(200)	Tiện nghi của phòng
6	MoTa	nvarchar(1000)	Mô tả phòng
7	LoaiGiuong	int	Loại giường
8	<u>IdKhachSan</u>	int	Khóa ngoại

### 3-7. Bảng ThanhPho.

STT	Thuộc tính	Kiểu	Ghi chú
1	<u>Id</u>	int	Khóa chính
2	Ten	nvarchar(100)	Tên thành phố
3	MoTa	nvarchar(200)	Mô tả về thành phố
4	UrlHinhAnh	varchar(200)	Hình ảnh thành phố

## CHƯƠNG 4. KẾT QUẢ THỰC NGHIỆM

### 4.1. Các thành phần chức năng của trang web

- **Admin** : đăng ký, đăng nhập, quản lý khách sạn thêm xóa sửa
- **Người dùng** : đăng ký, đăng nhập, tìm kiếm, đặt phòng

### 4.2 Giao diện trực quan.

#### 4.2.1. Giao diện quản lý.

Hình 3-1 giao diện admin quản lý tài khoản.

Chào admin,

Menu

Quản Trị

▼ Danh Mục

Tài Khoản

Thành Phố

Loại Khách Sạn

Khách Sạn

Phòng

▼ Người Dùng

Tài Khoản

Đánh giá













Đăng Xuất

(1 of 1) << 1 >>

Thêm Mới

Tên Tài Khoản	Mật Khẩu	Họ Tên	Giới Tính	Số Điện Thoại	Email	Là Admin	Thao Tác
a	1	Nam	Nam	0234565422	abc@gmail.com	Không	<div><div></div><div></div></div>
admin	111	admin	Nam	034356xxx09	buivanbaoth@gmail.com	Có	<div><div></div><div></div></div>
User	123456	Người Dùng	Nữ	0358763412	user@gmail.com	Không	<div><div></div><div></div></div>

Hình 3-2 giao diện admin quản lý thành phố.

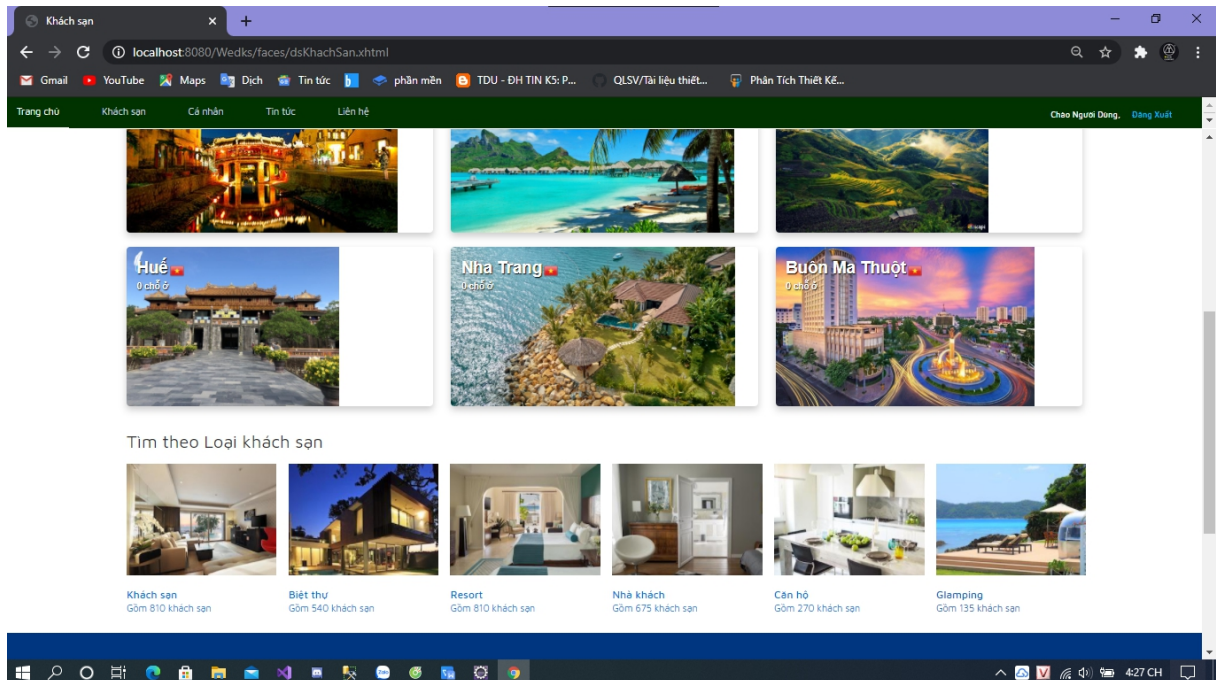
Menu	(1 of 2) << < 1 2 > >> + Thêm Mới			
Quản Trị	Id	Tên Thành Phố	Mô Tả	Thao Tác
▼ Danh Mục	1	Hà Nội	Thành Phố Hà Nội	 
☰ Tài Khoản	2	TP. Hồ Chí Minh	Thành Phố Hồ Chí Minh	 
☰ Thành Phố	3	Đà Nẵng	Thành Phố Đà Nẵng	 
☰ Loại Khách Sạn	4	Đà Lạt	Thành Phố Đà Lạt	 
☰ Khách Sạn	5	Hội An	Thành Phố Hội An	 
☰ Phòng	6	Phú Quốc	Thành Phố Phú Quốc	 
▼ Người Dùng				
☰ Tài Khoản				
☰ Đánh giá				
☰ Đăng Xuất				

Hình 3-3 giao diện admin quản lý loại khách sạn.

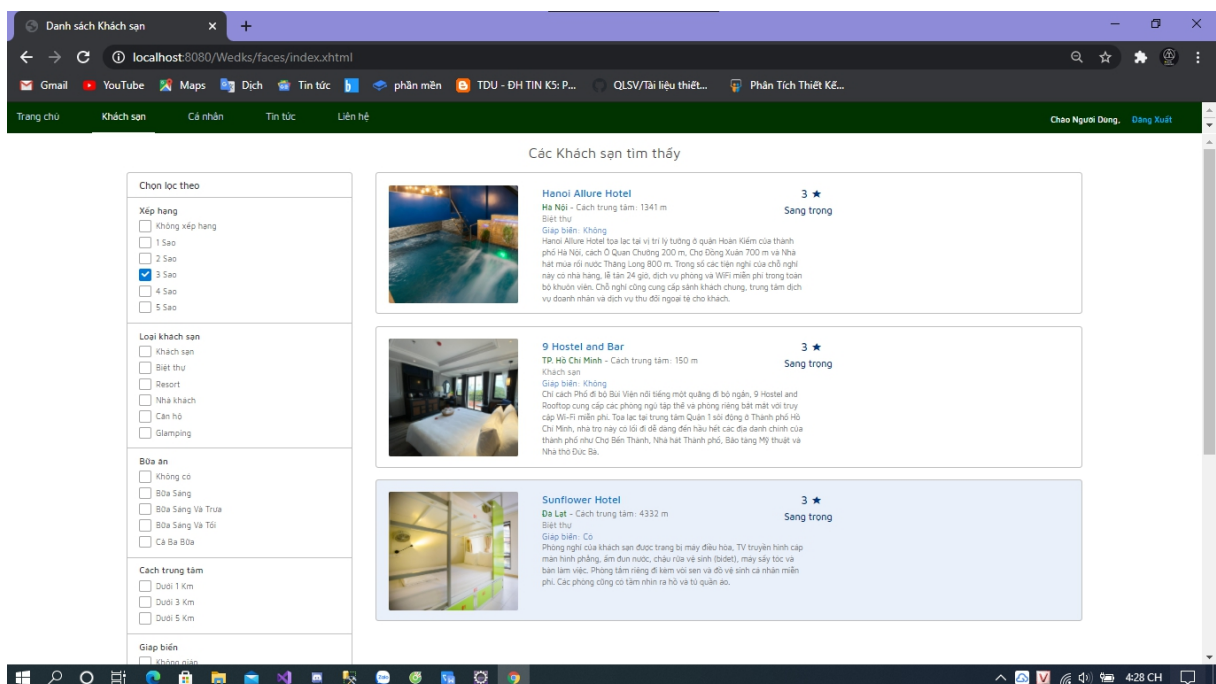


## 4.2.2. Giao diện khách truy cập.

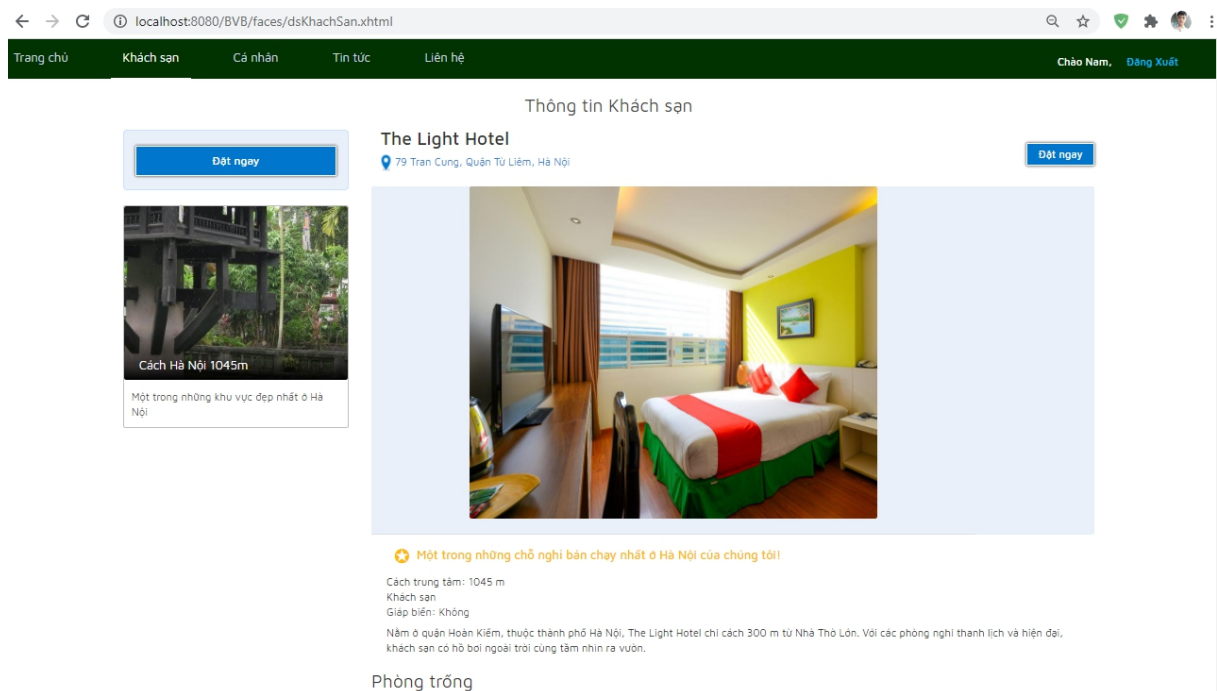
Hình 3-6 giao diện khách truy cập trang chủ.



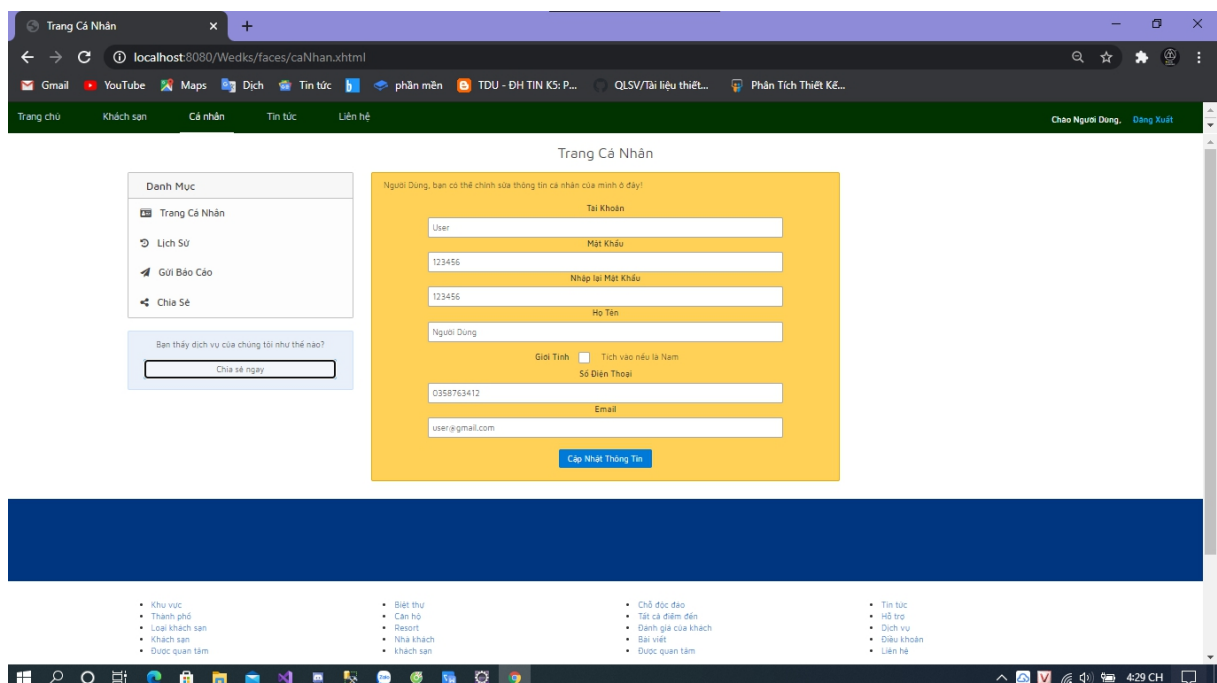
Hình 3-7 giao diện khách truy cập khách sạn để đặt phòng.



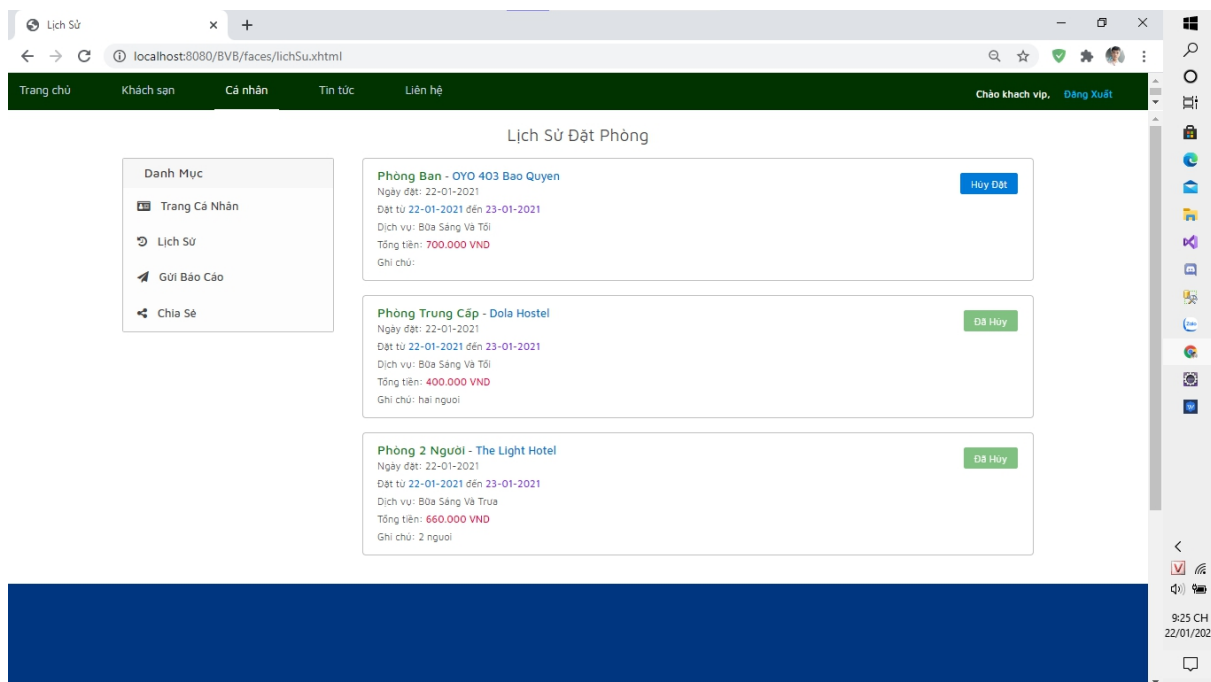
Hình 3-8 giao diện khách truy cập thông tin khách sạn có chi tiết mô tả để đặt phòng.



Hình 3-9 giao diện khách truy cập thông tin cá nhân.



Hình 3-10 giao diện khách truy cập lịch sử đặt phòng.





## **CHƯƠNG 5. KẾT LUẬN VÀ KIẾN NGHỊ**

### **5.1. Kết luận chung**

Về cơ bản, trang web quản lý khách sạn đã giúp người dùng liên lạc và trao đổi với khách sạn một cách tiện lợi, giúp người dùng đặt phòng cũng như tìm kiếm phòng một cách nhanh nhất.

Xây dựng các trang web nói chung và trang web quản lý khách sạn nói riêng không chỉ là việc xây dựng đơn thuần mà phải đòi hỏi người lập trình lên ý tưởng cho trang web thật kĩ càng, phù hợp, lựa chọn đúng định hướng cho trang web, phải thiết kế như thế nào để phù hợp với người dùng.

Qua việc nghiên cứu xây dựng đề tài này, chúng em đã phần nào củng cố được các kiến thức về lập trình, giúp chúng em tiếp thu nhiều kiến thức mới và nắm được cách làm thế nào để xây dựng được một trang web.

### **5.2. Hướng phát triển và mở rộng đồ án**

Như trên đã trình bày, hệ thống đã xây dựng còn khá nhiều hạn chế cả về giao diện lẫn chức năng, trong đó là khả năng liên kết với các nhà cung cấp dịch vụ còn khá ít. Trong tương lai, có thể mở rộng ra liên kết với các khách sạn, hãng không không và đồng thời cũng phối hợp với các dịch vụ liên quan khác như: thuê nhà, thuê xe, kết hợp với các tour du lịch... Nếu làm được như thế thì website sẽ có thể hoàn thiện hơn.

Trong đồ án này, chúng em chỉ mới phân tích và xây dựng trang web đơn giản, cần phải phát triển và làm rõ thêm:

## TÀI LIỆU THAM KHẢO

- [1] <https://shareeverything.com/lap-trinh/tim-hieu-spring-mvc-va-spring-boot-trong-java-web/>
- [2] <https://stackjava.com/spring/spring-mvc-phan-1-kien-truc-cua-spring-mvc.html>
- [3] <https://viettuts.vn/jsp>
- [4] <https://niithanoi.edu.vn/jsp-la-gi.html>
- [5] <https://shareprogramming.net/jdbc-la-gi-khong-can-hoc-jdbc-dau/>
- [6] <https://viblo.asia/p/websocket-la-gi-Ljy5VxkbZra>