



24/01/2021

# Ứng dụng chia sẻ nhà trọ

Framework Struts 2



Huỳnh Tấn Phát  
QUY NHƠN UNIVERSITY

TRƯỜNG ĐẠI HỌC QUY NHƠN  
KHOA CÔNG NGHỆ THÔNG TIN

---



**BÁO CÁO BÀI TẬP**  
**ĐỀ TÀI: ỨNG DỤNG CHIA SẺ NHÀ TRỢ**

Giảng viên hướng dẫn : ***Nguyễn Thanh Bình***

Sinh viên thực hiện: ***Huỳnh Tấn Phát***

Lớp: ***CNTT-K40A***

Quy Nhơn, 24/01/2021

# Mục lục

<b>I. Framework Struts 2 .....</b>	<b>1</b>
1. Khái niệm.....	1
2. Lịch sử phát triển .....	1
3. Thuật ngữ .....	1
4. Thành phần.....	2
5. Kiến trúc .....	4
6. Quy trình Struts2 .....	5
7. Tính năng Struts2 .....	5
8. Nhược điểm của Struts2 .....	6
<b>II. Ứng dụng chia sẻ nhà trọ .....</b>	<b>7</b>
1. Giới thiệu .....	7
2. Lợi ích thực tế .....	7
3. Biểu đồ ca sử dụng .....	8
4. Biểu đồ luồng (Flowchart).....	9
5. Cơ sở dữ liệu.....	10
6. Giao diện ứng dụng.....	11
a) Giao diện Login (Đăng nhập).....	11
b) Giao diện Register (Đăng kí).....	11
c) Giao diện User's Dashboard (Trang chủ của người dùng) .....	12
d) Giao diện Profile (Admin & User).....	12
e) Giao diện User's Listrooms (Danh sách phòng của người dùng) .....	13
f) Giao diện Shares Rooms (Chia sẻ phòng) (Admin & User).....	13
g) Giao diện Admin Dashboard (Trang chủ người quản trị).....	14
h) Giao diện Pending Rooms (Danh sách phòng chờ duyệt).....	14
i) Giao diện User Management (Quản lý người dùng).....	15
j) Giao diện Room's Details (Thông tin chi tiết của phòng).....	15

k) Giao diện 404 .....	16
7. Cấu trúc dự án .....	16
8. Kỹ thuật & công cụ.....	18
a) Interceptor .....	18
b) Action With The Name “*” .....	18
c) Session.....	19
d) Iterator & Property .....	20
e) DataTables.....	21
f) Classess .....	23
g) Two Different Way Import File .....	25
h) Packet Management.....	25
9. Tài liệu tham khảo .....	26

# I. Framework Struts 2

## 1. Khái niệm

**Framework** là các đoạn code đã được viết sẵn, cấu thành nên một bộ khung và các thư viện lập trình được đóng gói. Chúng cung cấp các tính năng có sẵn như mô hình, API và các yếu tố khác để tối giản cho việc phát triển các ứng dụng web phong phú, năng động.

**Struts 2 Framework** là Framework hỗ trợ thiết kế và xây dựng ứng dụng web theo mô hình MVC 2 thông qua việc cung cấp cho người dùng hàng loạt các thư viện và tiện ích.

## 2. Lịch sử phát triển

Ban đầu được phát triển bởi lập trình viên kiêm tác giả Craig R. McClanahan, sau đó được Apache Software Foundation tiếp quản vào năm 2002. Struts đã cung cấp một khuôn khổ tuyệt vời để phát triển ứng dụng dễ dàng bằng cách tổ chức JSP và Servlet dựa trên các định dạng HTML và mã Java. Struts1 với tất cả các công nghệ và gói Java tiêu chuẩn của Jakarta hỗ trợ tạo ra một môi trường phát triển có thể mở rộng. Tuy nhiên, với nhu cầu ngày càng tăng của ứng dụng web, Struts 1 không đứng vững và cần phải thay đổi theo nhu cầu.

Struts2 chứa các tính năng kết hợp của các dự án Struts và WebWork 2 ủng hộ ứng dụng cấp cao hơn bằng cách sử dụng kiến trúc của WebWork 2 với các tính năng bao gồm khung plugin, API mới, thẻ Ajax, v.v. Vì vậy, cộng đồng Struts và nhóm WebWork đã tập hợp một số tính năng đặc biệt trong WebWork 2 để làm cho nó tiến bộ hơn trong thế giới Nguồn mở. Sau đó, tên của WebWork 2 đã đổi thành Struts2. Do đó, Struts 2 là một khuôn khổ năng động, có thể mở rộng để phát triển ứng dụng hoàn chỉnh từ xây dựng, triển khai và bảo trì.

## 3. Thuật ngữ

### **MVC = Model View Controller**

+ **Model**: Là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Bộ phận này là một cầu nối giữa 2 thành phần bên dưới là View và Controller.

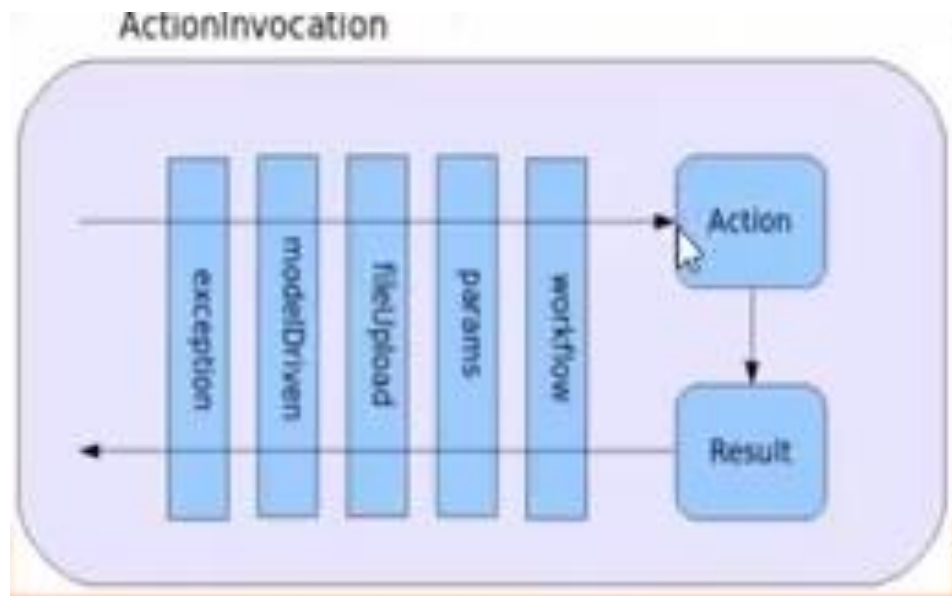
+ **View**: Đây là phần giao diện (theme) dành cho người sử dụng.

+ **Controller**: là phần xử lý logic, xử lý các yêu cầu người dùng đưa đến thông qua View.

- **OGNL**: là viết tắt của Object Graph Navigation Language, là một ngôn ngữ diễn đạt mạnh mẽ được sử dụng để tham chiếu và thao tác dữ liệu trên ValueStack, giúp truyền tải và chuyển đổi kiểu của dữ liệu.
- **POJO hay POJOs**: là từ viết tắt của cụm từ “Plain Old Java Object” có nghĩa là “Các đối tượng Java thuần túy” thường được dùng để chỉ các chỉ những object Java bình thường, không implement hay extend class nào khác. Đơn giản bao gồm các thuộc tính và các phương thức.
- **JSP**: là viết tắt của JavaServer Pages là một công nghệ để phát triển các trang web động. JSP giúp các nhà phát triển chèn java code vào các trang HTML bằng cách sử dụng các thẻ JSP đặc biệt.

#### 4. Thành phần

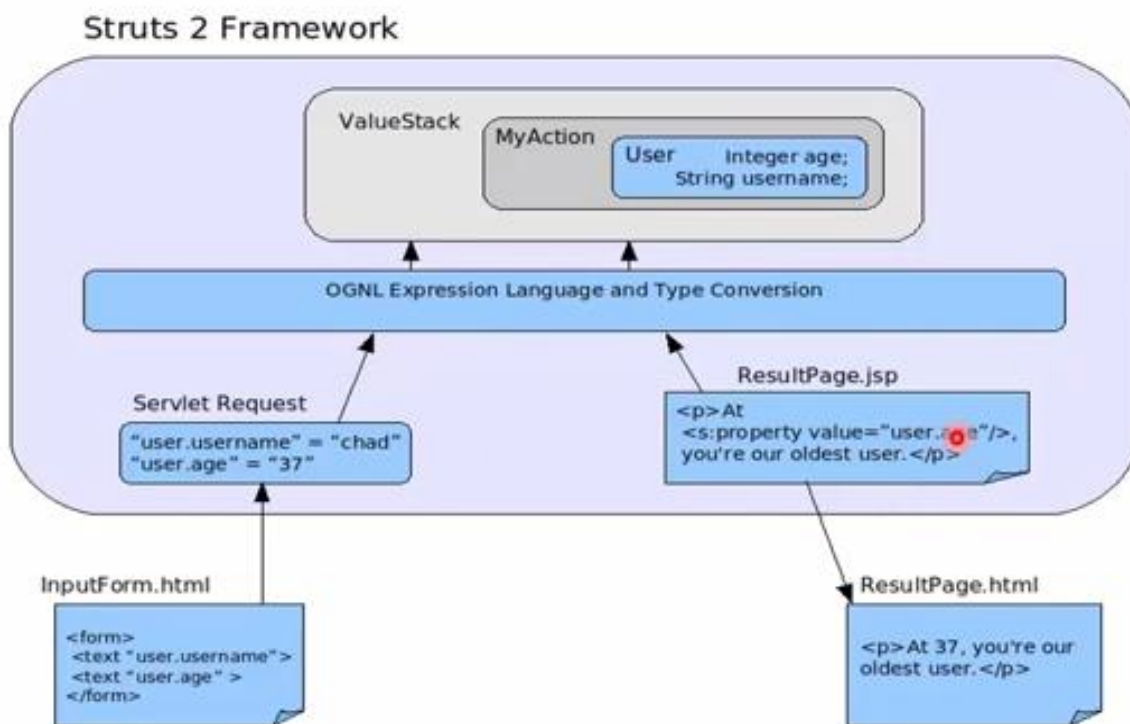
- **Action**: khi client muốn request đến server 1 thông tin gì đó thì nó phải request đến 1 action và tùy thuộc vào action sẽ đưa ra view tương ứng. Ở đây view đóng vai trò xử lý mọi thao tác liên quan đến tương tác với người dùng.



- **Interceptor**: đóng vai trò và hoạt động gần tương tự như một filter (chặn và sửa các yêu cầu (requests) hoặc các phản hồi (responses) từ người dùng hoặc máy chủ), Interceptor được gọi đến trước khi thông tin đi đến một actions.
- **Value Stack**: là nơi chứa tất cả các thông tin mà action, view cần như session request, hay các thuộc tính trong action, model...

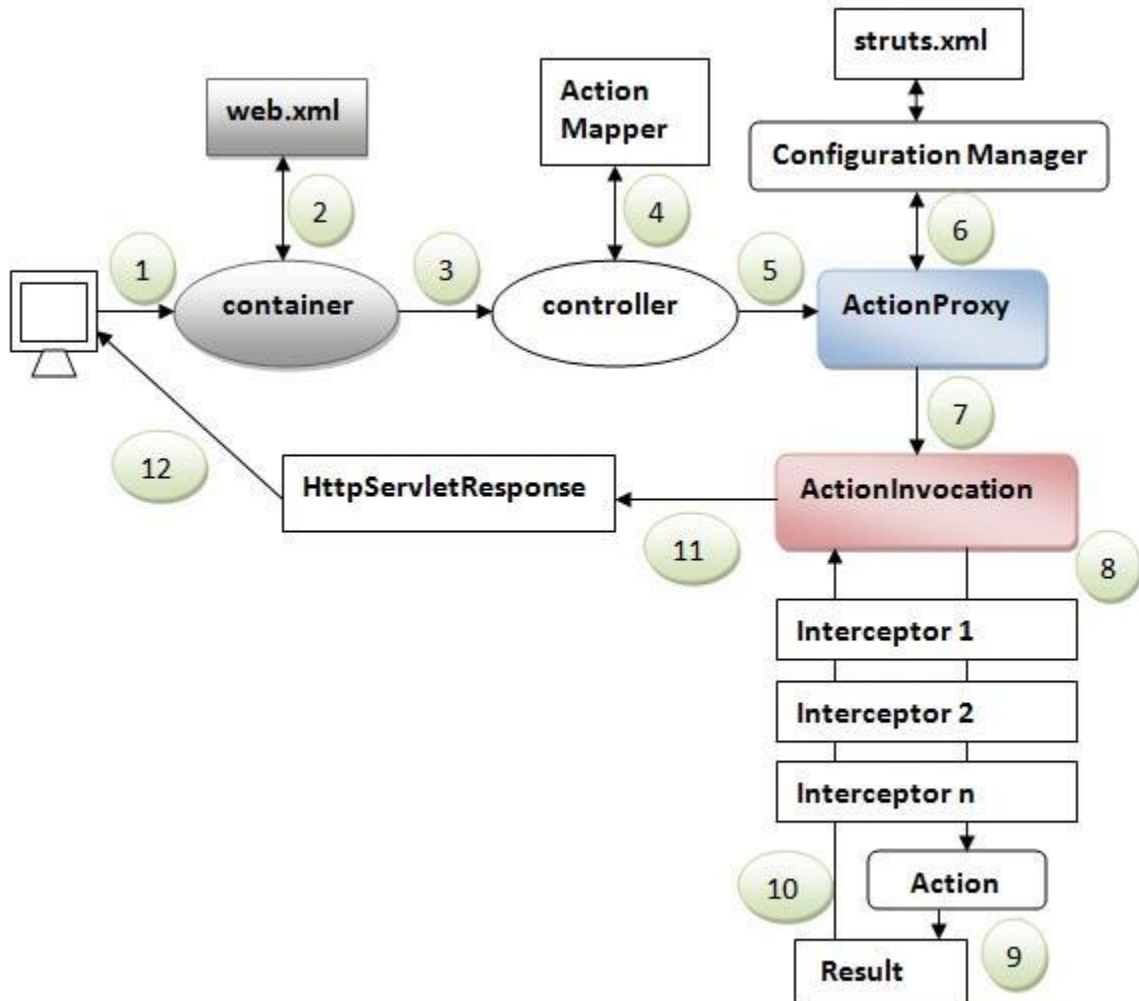
SN	Objects & Description
1	<b>Temporary Objects</b> There are various temporary objects which are created during execution of a page. For example the current iteration value for a collection being looped over in a JSP tag.
2	<b>The Model Object</b> If you are using model objects in your struts application, the current model object is placed before the action on the value stack
3	<b>The Action Object</b> This will be the current action object which is being executed.
4	<b>Named Objects</b> These objects include #application, #session, #request, #attr and #parameters and refer to the corresponding servlet scopes

- **Object-Graph Navigation (OGNL):** nói đơn giản thì đây là một ngôn ngữ mạnh mẽ để giúp chúng ta có thể lấy dữ liệu.



- **Type Conversion và Validation:** Struts2 hỗ trợ việc validation cho client và server, nó cung cấp và thực hiện khai báo qua tập tin xml.

## 5. Kiến trúc

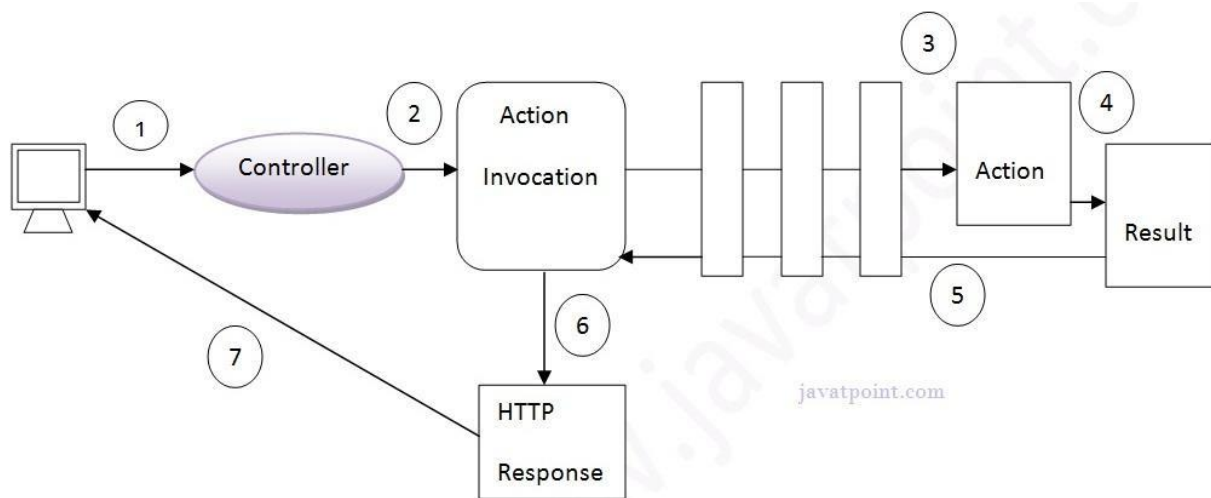


1. Người dùng gửi yêu cầu hành động.
2. Vùng chứa ánh xạ yêu cầu trong tệp web.xml và lấy tên lớp của bộ điều khiển.
3. Vùng chứa gọi bộ điều khiển (StrutsPrepareAndExecuteFilter hoặc FilterDispatcher).
4. Bộ điều khiển lấy thông tin cho hành động từ ActionMapper.
5. Bộ điều khiển gọi ActionProxy.
6. ActionProxy lấy thông tin về hành động và ngăn xếp bộ chặn từ trình quản lý cấu hình lấy thông tin từ tệp struts.xml.
7. ActionProxy chuyển tiếp yêu cầu tới ActionInvocation.



8. ActionInvocation gọi từng bộ đánh chặn và hành động.
9. Một kết quả được tạo ra.
10. Kết quả được gửi trở lại ActionInvocation.
11. HttpServletResponse được tạo.
12. Phản hồi được gửi đến người dùng.

## 6. Quy trình Struts2



1. Người dùng gửi yêu cầu hành động.
2. Bộ điều khiển gọi ActionInvocation.
3. ActionInvocation gọi từng bộ đánh chặn và hành động.
4. Một kết quả được tạo ra.
5. Kết quả được gửi trở lại ActionInvocation.
6. HttpServletResponse được tạo.
7. Phản hồi được gửi đến người dùng.

## 7. Tính năng Struts2

**Các POJO form và POJO action:** Struts2 đã loại bỏ các Form Action - một phần không thể tách rời của Struts framework. Với Struts2, bạn có thể sử dụng bất kỳ POJO nào để nhận dữ liệu từ form. Tương tự như vậy, với Struts2 bạn có thể xem bất kỳ POJO nào làm lớp Action.

**Hỗ trợ thẻ:** Struts2 đã cải tiến các thẻ form và các thẻ mới nhằm giúp các nhà phát triển viết mã ít hơn

**Hỗ trợ AJAX:** Struts2 đã công nhận sự tiếp quản của các công nghệ Web2.0 và đã tích hợp hỗ trợ AJAX vào sản phẩm bằng cách tạo các thẻ AJAX có chức năng rất giống với các thẻ Struts2 tiêu chuẩn.

**Tích hợp dễ dàng:** Việc tích hợp Struts2 với các framework khác như Spring, Tiles và SiteMesh giờ đây đã trở nên dễ dàng hơn.

**Hỗ trợ Template:** Hỗ trợ tạo ra các view bằng việc sử dụng các template.

**Hỗ trợ Plugin:** Các hành vi của core Struts2 có thể được cải tiến bằng cách sử dụng các plugin. Hiện nay có khá nhiều plugin có sẵn cho Struts2.

**Profiling:** Struts2 cung cấp tích hợp profiling để gỡ lỗi ứng dụng. Ngoài ra, Struts cũng cung cấp gỡ lỗi được tích hợp với sự trợ giúp của công cụ gỡ lỗi được xây dựng bên trong.

**Dễ dàng sửa đổi các thẻ** Tag markups trong Struts2 có thể được tinh chỉnh bằng cách sử dụng các mẫu Freemarker. Điều này không yêu cầu kiến thức JSP hoặc java. Bạn có kiến thức cơ bản về HTML, XML và CSS đủ để sửa đổi các thẻ.

**Cấu hình ít hơn:** Struts2 giúp bạn cấu hình ít hơn với sự trợ giúp của việc sử dụng các giá trị mặc định cho các cài đặt khác nhau. Bạn không cần phải cấu hình một cái gì đó trừ khi bạn muốn thiết lập khác các thiết lập mặc định được thiết lập bởi Struts2.

**Các công nghệ View:** Struts2 có một sự hỗ trợ tuyệt vời cho nhiều lựa chọn view (JSP, Freemarker, Velocity và XSLT)

## **8. Nhược điểm của Struts2**

**Khả năng tương thích:** Struts 2 hoàn toàn khác với Struts 1. Vì vậy, rất khó để thực hiện việc di chuyển các ứng dụng từ Struts 1 sang Struts 2.

**Tài liệu giới hạn:** Tài liệu giới hạn có sẵn cho Struts 2. Ngoài ra, người dùng mới cảm thấy khó hiểu các khái niệm của nó do tài liệu được quản lý kém bởi Apache.

## **II. Ứng dụng chia sẻ nhà trọ**

### **1. Giới thiệu**

Tìm phòng trọ thực sự là quá trình khó khăn và mất nhiều thời gian đối với những bạn người mới lên thành phố sinh sống. Không những khó khăn trong việc tìm kiếm mà việc chọn được một nhà trọ tốt, an toàn cùng chi phí hợp lý cũng làm nhiều người đau đầu. Đó là tiền đề ra đời của ứng dụng chia sẻ nhà trọ. Ứng dụng chia sẻ phòng trọ đem lại sự tiện lợi, nhanh chóng, tiết kiệm thời gian đồng thời hạn chế rủi ro với vài thao tác đơn giản.

### **2. Lợi ích thực tế**

Ứng dụng chia sẻ phòng trọ giúp tiết kiệm được lượng lớn thời gian cho người có nhu cầu tìm trọ. Các hoạt động truyền thống thường làm mất rất nhiều thời gian và khiến người đi tìm trọ gặp rất nhiều khó khăn. Ngày nay, với công nghệ hiện đại phát triển, ứng dụng chia sẻ nhà trọ ra đời giúp rút ngắn thời gian, hiệu quả làm việc cao.

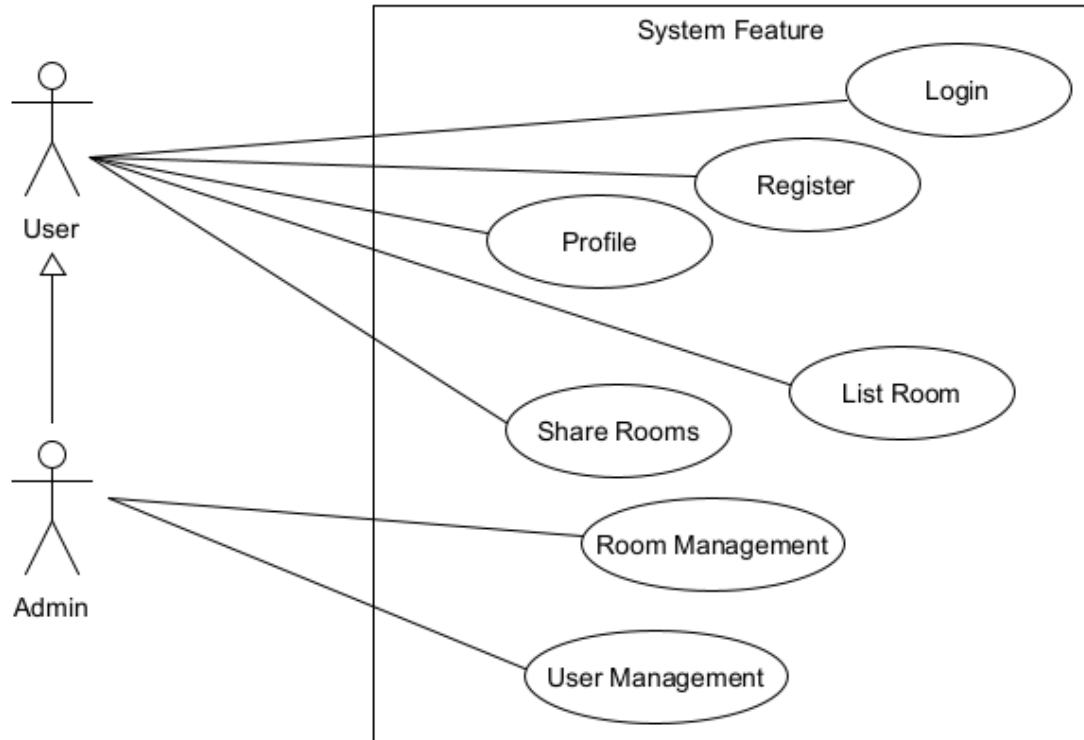
Cho phép bạn chủ động trong việc cập nhật thông tin mọi lúc mọi nơi tiện lợi trên các thiết bị có kết nối mạng với một trình duyệt web.

Người dùng có thể hạn chế được nhiều rủi ro khi sử dụng phần mềm chia sẻ phòng trọ. Không gian sống cũng vì thế mà được đảm bảo, mang lại cho khách hàng cảm giác an toàn trong quá trình sinh sống và làm việc. Người đi tìm phòng trọ sẽ chủ động nắm được tình hình về phòng qua các thông tin và hình ảnh trên ứng dụng.

Các bạn cũng có thể yên tâm vì thông tin trên các ứng dụng tìm phòng trọ được cập nhật thường xuyên và được kiểm duyệt. Ngoài ra, các chủ cho thuê phòng trọ cũng có thể đăng thông tin trên các ứng dụng tìm phòng trọ tại mục đăng tin hoàn toàn miễn phí.

Với giao diện được thiết kế đơn giản, dễ sử dụng, bạn không phải mất quá nhiều thời gian để tìm hiểu cách sử dụng. Từ những thông tin được cập nhật, bạn cũng có thể lựa chọn các tùy chỉnh như mức giá, diện tích, khu vực phù hợp với nhu cầu của bạn.

### 3. Biểu đồ ca sử dụng



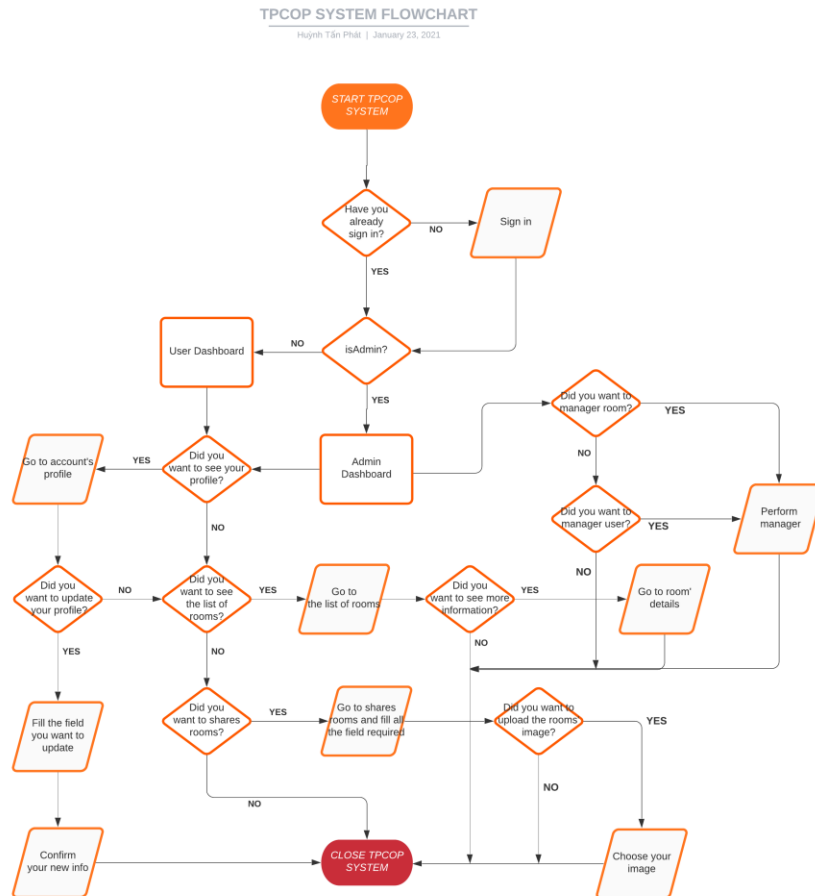
Nhìn sơ qua biểu đồ này ta có thể thấy người dùng (user) có thể thực hiện một số chức năng cơ bản như:

- ✓ Đăng nhập
- ✓ Đăng kí
- ✓ Xem thông tin tài khoản
- ✓ Xem danh sách phòng trọ

Còn người quản trị (admin) thì cũng có thể thực hiện những chức năng mà người dùng (user) có thể thực hiện và bổ sung thêm hai chức năng là:

- ✓ Quản lý phòng trọ
- ✓ Quản lý người dùng

## 4. Biểu đồ luồng (Flowchart)



Biểu đồ luồng mô tả luồng hoạt động của hệ thống từ khi truy cập cho đến khi kết thúc, qua đó cho thấy cái nhìn tổng quan để thực hiện lập trình (code). Ví dụ: với phiên làm việc là người dùng thì có thể thực hiện xem và sửa thông tin tài khoản (profile), xem danh sách phòng trọ đang có và được người quản trị (admin) duyệt và người dùng cũng có thể đăng bài chia sẻ phòng trọ qua chức năng chia sẻ phòng (shares rooms). Ngoài ra biểu đồ cũng cho thấy những điều kiện cần thiết để thực thi chức năng phía sau.

## 5. Cơ sở dữ liệu

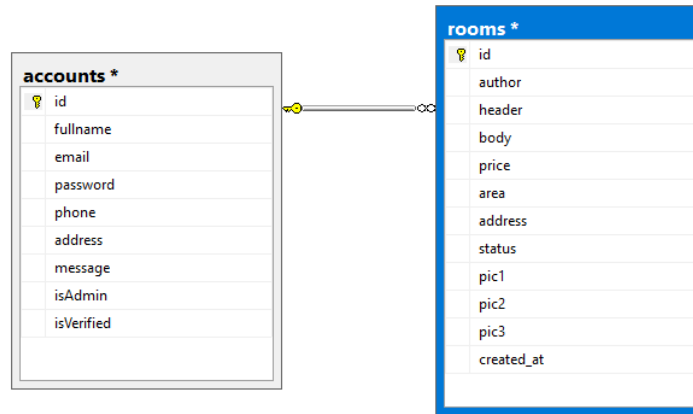


TABLE: **accounts**

Collumn Name	Data Type	Allow null	isUnique
id	int		True
fullname	nvarchar(30)		
email	varchar(50)		True
password	varchar(50)		
phone	varchar(50)	True	
address	nvarchar(200)	True	
message	nvarchar(200)	True	
isAdmin	varchar(1)		
isVerified	varchar(1)		

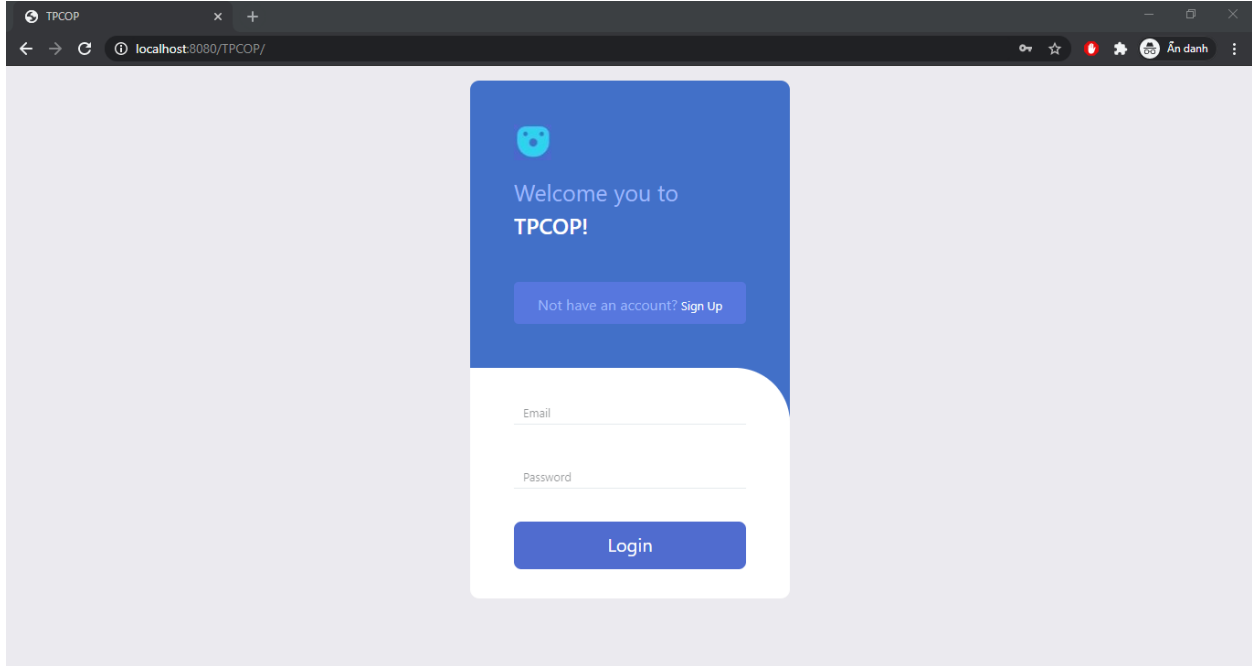
TABLE: **rooms**

Collumn Name	Data Type	Allow null	isUnique
id	int		True
author	nvarchar(30)		
header	nvarchar(150)		
body	nvarchar(1000)		
price	varchar(50)	True	
area	varchar(50)	True	
address	nvarchar(300)	True	
status	varchar(20)	True	
pic1	nvarchar(500)	True	
pic2	nvarchar(500)	True	
pic3	nvarchar(500)	True	
created_at	datetime		True

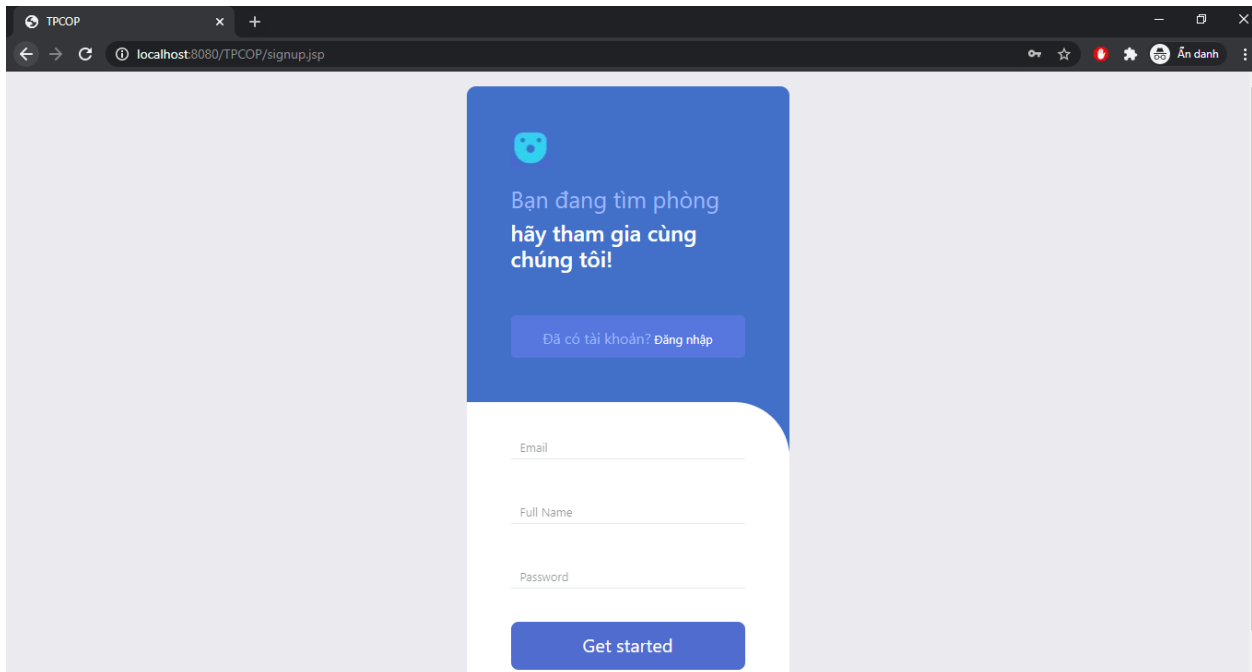
## 6. Giao diện ứng dụng

Ứng dụng sử dụng các mẫu thiết kế mã nguồn html, css, và javascript cụ thể là **NICE ADMIN** mã nguồn mở có sẵn trên internet, mã nguồn xem ở mục 9.

### a) Giao diện Login (Đăng nhập)



### b) Giao diện Register (Đăng kí)



### c) Giao diện User's Dashboard (Trang chủ của người dùng)

The screenshot shows the 'User Dashboard' of the 'NICE ADMIN' system. The left sidebar contains navigation links: Dashboard, Profile, List Rooms, Shares Rooms, and LogOut. The main content area has a search bar and a user profile 'John Barack'. Below this is a 'Dashboard' section with a welcome message 'Welcome back John Barack'. A 'File export' section offers buttons for Copy, CSV, Excel, PDF, and Print. A table displays room data with columns for #, Author, Title, Area (m²), Price (Millions), and Status. The table contains three rows of data, each with an 'Open' button in the Status column.

#	Author	Title	Area (m²)	Price (Millions)	Status
5	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	19	0.7	<button>Open</button>
7	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	21	2	<button>Open</button>
9	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	100	10	<button>Open</button>

### d) Giao diện Profile (Admin & User)

The screenshot shows the 'Profile' page of the 'NICE ADMIN' system. The left sidebar is the same as the dashboard. The main content area has a search bar and a user profile 'John Barack'. Below this is a 'Profile' section with a user profile card and a form to update the profile. The profile card shows the user's name, title, and contact information. The form includes fields for Full Name, Email, Password, Phone No, and a Message field, with an 'Update Profile' button at the bottom.

**Profile Card:**

- John Barack**  
Accounts Manager Amix corp
- 254** **54**
- Email address: test@gmail.com
- Phone: 0967437571
- Address: Quy Nhơn
- Social Profile

**Update Profile Form:**

- Full Name: John Barack
- Email: test@gmail.com
- Password: .....
- Phone No: 0967437571
- Message: [Text Area]
- Update Profile



### e) Giao diện User's Listrooms (Danh sách phòng của người dùng)

Rooms

File export

Copy CSV Excel PDF Print

Search:

#	Author	Title	Area (m <sup>2</sup> )	Price (Millions)	Status	Act
5	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	19	0.7	Open	
7	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	21	2	Open	
9	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	100	10	Open	
#	Author	Title	Area (m <sup>2</sup> )	Price (Millions)	Status	Act

Showing 1 to 3 of 3 entries

Previous 1 Next

### f) Giao diện Shares Rooms (Chia sẻ phòng) (Amin & User)

Shares Rooms

Title

Prices

Areas

Address

Message

Images

Chon tệp Không có tệp nào được chọn

Cancel Share Rooms

### g) Giao diện Admin Dashboard (Trang chủ người quản trị)

Admin Dashboard

localhost:8080/TPCOP/admin/golIndex.action

NICE ADMIN

Dashboard

Profile

Pending Rooms

Shares Rooms

User Management

LogOut

Search

Huỳnh Tấn Phát

Dashboard

Welcome back Huỳnh Tấn Phát

New Clients 12

New Request 2

Request Accepted 3

Request Closed 0

File export

Copy CSV Excel PDF Print

Search:

#	Author	Title	Area (m <sup>2</sup> )	Price (Millions)	Status
5	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	19	0.7	Open
6	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	19	0.7	Pending

### h) Giao diện Pending Rooms (Danh sách phòng chờ duyệt)

Lists Rooms

localhost:8080/TPCOP/admin/goRooms

NICE ADMIN

Dashboard

Profile

Pending Rooms

Shares Rooms

User Management

LogOut

Search

Huỳnh Tấn Phát

Rooms

File export

Copy CSV Excel PDF Print

Search:

#	Author	Title	Area (m <sup>2</sup> )	Price (Millions)	Status	Act
6	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	19	0.7	Pending	Q X ✓
8	test@gmail.com	Ở ghép máy lạnh, máy giặt, tắm nước nóng -700k Phạm Hùng	16	1	Pending	Q X ✓

Showing 1 to 2 of 2 entries

Previous 1 Next

## i) Giao diện User Management (Quản lý người dùng)

The screenshot displays the 'User Management' page of the NICE ADMIN system. The page includes a sidebar with navigation links: Dashboard, Profile, Pending Rooms, Shares Rooms, User Management (selected), and LogOut. The main content area shows a 'User Management' header with a search bar and a 'File export' section with buttons for Copy, CSV, Excel, PDF, and Print. Below this is a table of users with columns: #, Fullname, Email, Phone No., Role, Status, and Act. The table contains 6 rows of user data.

#	Fullname	Email	Phone No.	Role	Status	Act
8	Lê Văn Tèo	admin1@gmail.com	0967537571	user	unverified	<a href="#">Q</a> <a href="#">S</a> <a href="#">E</a>
10	Huỳnh Tấn Phát	phatanthai@gmail.com	0967537571	admin	verified	<a href="#">Q</a> <a href="#">S</a> <a href="#">E</a>
11	Lê Văn Tèo	phatanthai1@gmail.com	0967537571	user	unverified	<a href="#">Q</a> <a href="#">S</a> <a href="#">E</a>
12	John Barack	test@gmail.com	0967437571	user	verified	<a href="#">Q</a> <a href="#">S</a> <a href="#">E</a>
13	Lê Văn Tèo	test1@gmail.com	0967537571	user	unverified	<a href="#">Q</a> <a href="#">S</a> <a href="#">E</a>

## j) Giao diện Room's Details (Thông tin chi tiết của phòng)

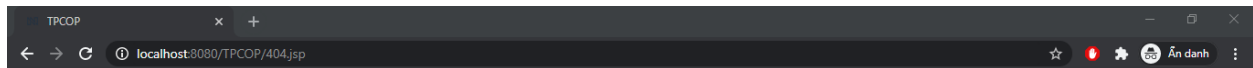
The screenshot displays the 'Room's Details' page of the NICE ADMIN system. The page shows a form with the following fields and values:

- Address: Đường Phạm Hùng, Phường 5, Quận 8, Hồ Chí Minh
- Sector: Hồ Chí Minh
- Type: Phòng trọ, nhà trọ
- PostID: #6
- Poster: John Barack
- Scope: Public
- Phone: 0967437571
- Update Date: 11/01/2021
- Area: 19m<sup>2</sup>
- Expires Date: unlimited
- Prices: 0.7 Millions Per Month

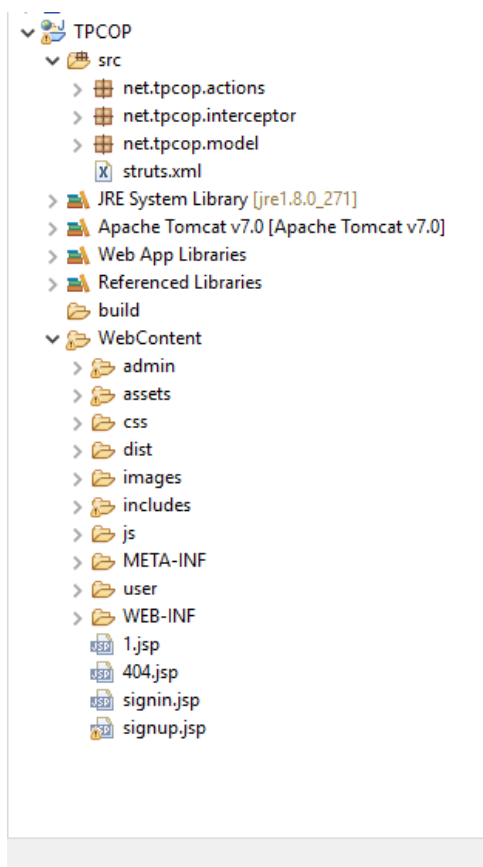
Below the form, there is a section titled 'Thông tin mô tả' (Description) containing a paragraph of text:

Cho thuê phòng trọ full nội thất, giờ giấc tự do: Máy lạnh, nem, cưa sô, bang công, wifi, máy giặt, thang máy, camera, WC,.... nhà gần chợ, ngân hàng, siêu thị, cưa hàng tiện lợi, trường cấp 1,2,3, thuận tiện đến các trường đại học, qua quán 5.6.1.7,... mất 5 ..15 phút. Phòng rộng rãi, thoáng mát, thích hợp NVVP, nhóm sinh viên, gia đình nhỏ. Giá từ :2tr\_4tr Liên hệ: 918AB TÀ QUANG BỬU, P5, quận 8. ĐT: 0908246366 gặp chỉ Chi

## k) Giao diện 404



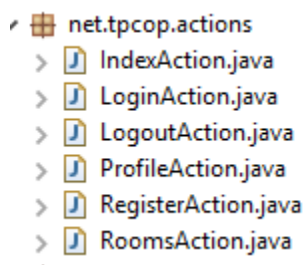
## 7. Cấu trúc dự án



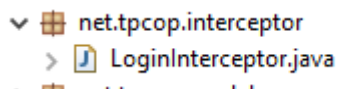
Toàn bộ tài nguyên ứng dụng sử dụng nằm trong thư mục **TPCOP**. Mã nguồn (source code) được đặt trong thư mục **src**, ngoài ra trong thư mục **src** còn chứa một file **struts.xml** đóng vai trò cấu hình, phân phối các kết quả xử lý. Các thư mục JRE System Library, Web App Libraries, Referenced Libraries chứa toàn bộ các thư viện mà ứng dụng sử dụng. Apache Tomcat v7.0 là thư mục chứa các thư viện của server để chạy ứng dụng (trong dự án này sử dụng Apache Tomcat tích hợp nên mới có thư mục Apache Tomcat v7.0, bình thường sẽ không có). Thư mục build chứa các file đã được biên dịch và sẵn sàng chạy trên server. Cuối cùng là thư mục WebContent chứa toàn bộ code giao diện (front-end) sử dụng trong dự án.

Để thuận tiện cho việc xử lý cũng như các thao tác liên quan truy xuất dữ liệu, sử dụng lại tài nguyên, toàn bộ mã nguồn của dự án được chia nhỏ ra làm 3 gói (packet): net.tpcop.actions, net.tpcop.interceptor và net.tpcop.model

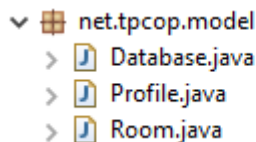
net.tpcop.actions: chứa các lớp Action mục đích để xử lý logic, dữ liệu từ các giao diện web ở mục 6.



net.tpcop.interceptor: chứa các lớp Interceptor để làm một số việc như: thông báo, kiểm tra session, kiểm tra lỗi,... Trong ứng dụng này sử dụng một lớp Interceptor để kiểm tra session.



net.tpcop.model: trong gói này chứa các lớp được phát hiện trong quá trình phát triển ứng dụng, được sử dụng xuyên suốt dự án.



## 8. Kỹ thuật & công cụ

### a) Interceptor

Như đã nói ở mục 7, trong dự án này sử dụng một interceptor để kiểm tra xem người dùng đã đăng nhập vào hệ thống hay chưa, nếu chưa đăng nhập thì chuyển hướng người dùng đến trang đăng nhập để sử dụng các chức năng bên trong.

```
<interceptors>
  <interceptor
    class="net.tpcop.interceptor.LoginInterceptor"
    name="loginInterceptor" />
  <interceptor-stack name="loginStack">
    <interceptor-ref name="loginInterceptor" />
    <interceptor-ref name="defaultStack" />
  </interceptor-stack>
</interceptors>
```

### b) Action With The Name "\*"

Sử dụng một Action với name "\*" để xử lý lỗi không tìm thấy trang ( đây chỉ là một mẹo nhanh, đúng ra là phải sử dụng interceptor), khi không có action nào gắn với request thì server sẽ trả về Action "\*", chuyển hướng đến trang 404.jsp.

```
<action name="*">
  <result type="redirect">
    <param name="location">404.jsp</param>
    <param name="namespace">/</param>
  </result>
</action>
```



PAGE NOT FOUND !

YOU SEEM TO BE TRYING TO FIND HIS WAY HOME

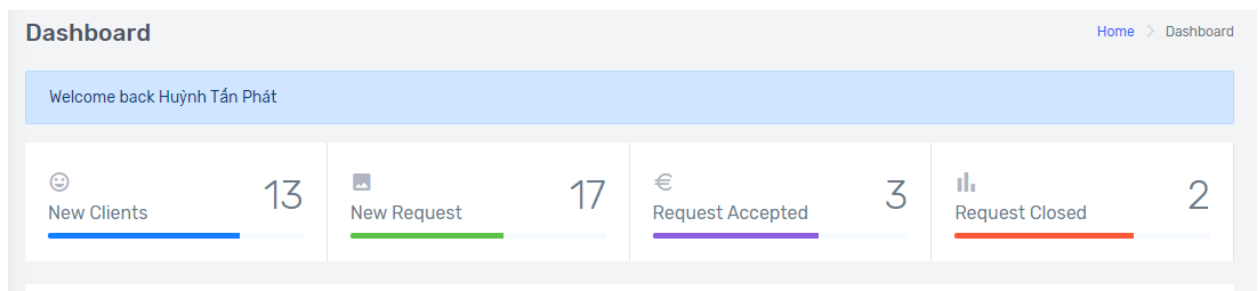
[Back to home](#)

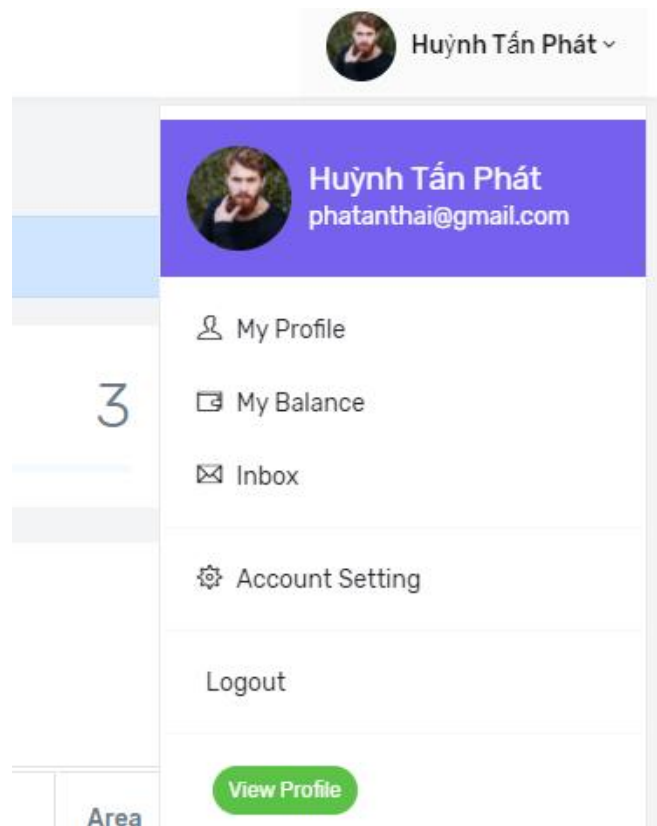
### c) Session

Session được sử dụng trong ứng dụng để xử lý nơi người dùng sẽ tương tác tiếp theo sau khi đăng nhập thành công, và các thao tác xử lý khác. Đồng thời một session cũng được sử dụng để đẩy thông báo tới người dùng. Cứ mỗi lần thông báo sẽ xóa session thông báo đi, còn session user sẽ được xóa khi người dùng đóng trình duyệt hoặc đăng xuất.

```
session.put("EMAIL", email);
session.put("USERNAME", rs.getString("fullname"));
session.put("VERIFIED", rs.getString("isVerified"));
session.put("ADMIN", rs.getString("isAdmin"));
session.put("NOTIFICTYPE", "1");
session.put("NOTIFICBODY", "Welcome back " + rs.getString("fullname"));
```

```
1 <s:if test='${#session.NOTIFICBODY != null}'>
2   <s:if test='${#session.NOTIFICTYPE == "1"}'>
3     <div class="alert alert-info">${NOTIFICBODY}</div>
4   </s:if>
5   <s:else>
6     <div class="alert alert-danger">${NOTIFICBODY}</div>
7   </s:else>
8   <%
9     request.getSession().removeAttribute("NOTIFICBODY");
10  %>
11 </s:if>
```





#### d) Iterator & Property

Để thể hiện các bảng trong cơ sở dữ liệu dưới góc nhìn của người dùng (Graphical User Interface), ta thực hiện bằng cách sử dụng hai cặp thẻ cơ bản do framework struts 2 cung cấp giúp lập trình viên vẽ (render) dữ liệu một cách dễ dàng nhất có thể đó là `s:iterator` và `s:property` như hình bên dưới:

```
<s:iterator value="datalist" var="room">
  <tr>
    <th scope="row"><s:property value="id" /></th>
    <td><s:property value="author" /></td>
    <td><s:property value="header" /></td>
    <td align="center"><s:property value="area" /></td>
    <td align="center"><s:property value="price" /></td>
    <td><s:if test='%{#room.status == "Open"}'%>
      <span class="Label label-success label-rounded">Open</span>
    </s:if> <s:elseif test='%{#room.status == "Closed"}'%>
      <span class="Label label-danger label-rounded">Closed</span>
    </s:elseif> <s:else>
      <span class="Label label-inverse label-rounded">Pending</span>
    </s:else></td>
    <td><span
      style="text-align: left; display: table-cell; min-width: 10px; padding-right: 10px;">
      <form action="cancelRequest" method="POST">
    </span> <span>
  </td>
</tr>
</s:iterator>
```



## e) DataTables

Để phân trang, định dạng, cũng như thêm các chức năng như tìm kiếm, sắp xếp cho các bảng được tạo ra bằng cách sử dụng `s:iterator`, ứng dụng đã sử dụng một thư viện mã nguồn mở tên DataTables:

```
<table id="file_export"
  class="table table-striped table-bordered display dataTable"
  role="grid" aria-describedby="file_export_info">
  <thead>
  <tbody>
    <s:iterator value="dataList" var="room">
      <tr>
        <th scope="row"><s:property value="id" /></th>
        <td><s:property value="author" /></td>
        <td><s:property value="header" /></td>
        <td align="center"><s:property value="area" /></td>
        <td align="center"><s:property value="price" /></td>
        <td><s:if test='%{#room.status == "Open"}'>
          <span class="label label-success label-rounded">Open</span>
        </s:if> <s:elseif test='%{#room.status == "Closed"}'>
          <span class="label label-danger label-rounded">Closed</span>
        </s:elseif> <s:else>
          <span class="label label-inverse label-rounded">Pending</span>
        </s:else></td>
        <td><span
          style="text-align: left; display: table-cell; min-width: 10px; padding: 5px;"
          <form action="cancelRequest" method="POST">
        </span> <span>
      </td>
    </tr>
  </s:iterator>
  </tbody>
  <tfoot>
</table>
```

Search:

# ↑↓	Author ↑↓	Title ↑↓	Area (m <sup>2</sup> ) ↑↓	Price (Millions) ↑↓	Status ↑↓
22	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH ,CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
23	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH ,CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
24	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH ,CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
25	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH ,CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
26	test@gmail.com	Phòng trọ có nội thất. Giờ giấc tự do. Giá: 2tr_3tr/tháng: ĐC:13/3, Bông Sao, p5, q8: ĐT:0903970791 gặp anh Lộc	12	3	Pending
#	Author	Title	Area (m <sup>2</sup> )	Price (Millions)	Status

Showing 1 to 5 of 5 entries

Previous

1

Next

Để thêm các tùy chọn báo cáo, xuất các bảng ra thành các file để người dùng xử lý sau đó, ứng dụng sử dụng thêm ba trình cắm (plug in) được lập trình sẵn thực hiện một loạt các thao tác xử lý chéo chéo trên DataTables:

```
24<script
25  src="https://cdn.datatables.net/buttons/1.5.1/js/dataTables.buttons.min.js"></script>
26<script
27  src="https://cdn.datatables.net/buttons/1.5.1/js/buttons.flash.min.js"></script>
28<script
29  src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
30<script
31  src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.32/pdfmake.min.js"></script>
32<script
33  src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.32/vfs_fonts.js"></script>
34<script
35  src="https://cdn.datatables.net/buttons/1.5.1/js/buttons.html5.min.js"></script>
36<script
37  src="https://cdn.datatables.net/buttons/1.5.1/js/buttons.print.min.js"></script>
38 <script src="../dist/js/pages/datatable/datatables-advanced.init.js"></script>
```

File export

Copy CSV Excel PDF Print

Search:

# ↑↓	Author ↑↓	Title ↑↓	Area (m <sup>2</sup> ) ↑↓	Price (Millions) ↑↓	Status ↑↓
22	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH .CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
23	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH .CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
24	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH .CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
25	phatanthai@gmail.com	CHO THUÊ PHÒNG MÁY LẠNH .CÓ BAN CÔNG THOÁNG MÁT	31	2	Pending
26	test@gmail.com	Phòng trọ có nội thất. Giờ giấc tự do. Giá: 2tr_3tr/tháng: ĐC:13/3, Bông Sao, p5, q8: ĐT:0903970791 gặp anh Lộc	12	3	Pending

## f) Classess

Phân tích ứng dụng, phát hiện các lớp tiềm năng được sử dụng với tần suất cao xuyên suốt dự án

- ▼ net.tpcop.model
  - ▼ Database.java
    - > Database
  - ▼ Profile.java
    - > Profile
  - ▼ Room.java
    - > Room
- ▼ net.tpcop.model
  - ▼ Database.java
    - ▼ Database
      - getConnection(String, String, String) : Connection
      - ▢ DB\_URL
      - ▢ PASSWORD
      - ▢ rs
      - ▢ USER\_NAME
      - Database()
      - execute(String) : boolean
      - executeQuery(String) : ResultSet
      - getConnection() : Connection

```

v Profile.java
  v Profile
    email
    fullname
    id
    isAdmin
    isVerified
    message
    password
    phone
    Profile()
    getEmail() : String
    getFullName() : String
    getId() : String
    getIsAdmin() : String
    getIsVerified() : String
    getMessage() : String
    getPassword() : String
    getPhone() : String
    setEmail(String) : void
    setFullName(String) : void
    setId(String) : void
    setIsAdmin(String) : void
    setIsVerified(String) : void
    setMessage(String) : void
    setPassword(String) : void
    setPhone(String) : void

v Room.java
  v Room
    address
    area
    author
    body
    date
    header
    id
    pic1
    pic2
    pic3
    price
    status
    Room()
    getAddress() : String
    getArea() : String
    getAuthor() : String
    getBody() : String
    getDate() : Date
    getHeader() : String
    getId() : String
    getPic1() : String
    getPic2() : String
    getPic3() : String
    getPrice() : String
    getStatus() : String
    setAddress(String) : void
    setArea(String) : void
    setAuthor(String) : void
    setBody(String) : void
    setDate(Date) : void
    setHeader(String) : void

```

## g) Two Different Way Import File

Sự khác nhau giữa hai cách include file trong jsp là:

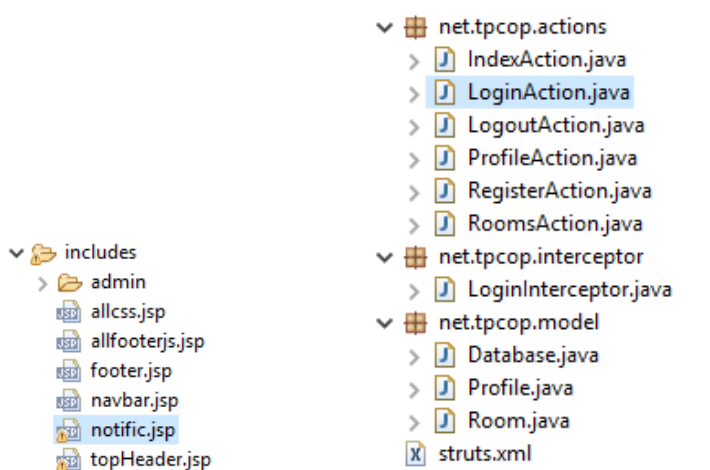
<code>&lt;%@include file="filepath"%&gt;</code>	<code>&lt;jsp:include page="filepath"/&gt;</code>
<p>Các câu lệnh trong filepath sẽ được thêm vào file jsp gốc rồi sau đó mới thực thi.</p> <p>⇒ Nên mọi file js, html, jsp, css,... sẽ thực thi đúng 1 lần</p>	<p>Các câu lệnh trong filepath sẽ được thực thi trong thời gian chạy (runtime)</p> <p>⇒ Số lần thực thi sẽ có thể tăng lên 2 vì hệ thống sẽ thực thi một lần ở runtime trước rồi sau đó mới thêm đoạn mã nguồn đó vào file gốc. Do đó một số đoạn mã javascript sẽ có thể thực thi một lần trong runtime rồi sau đó thực thi tiếp một lần nữa trong file gốc</p>

```
<!-- ===== -->
<jsp:include page="../includes/admin/navbar.jsp"></jsp:include>
<!-- ===== -->
```

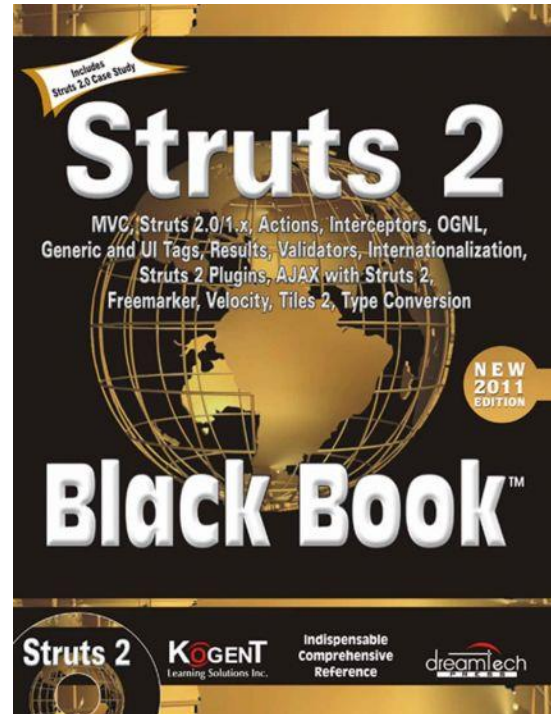
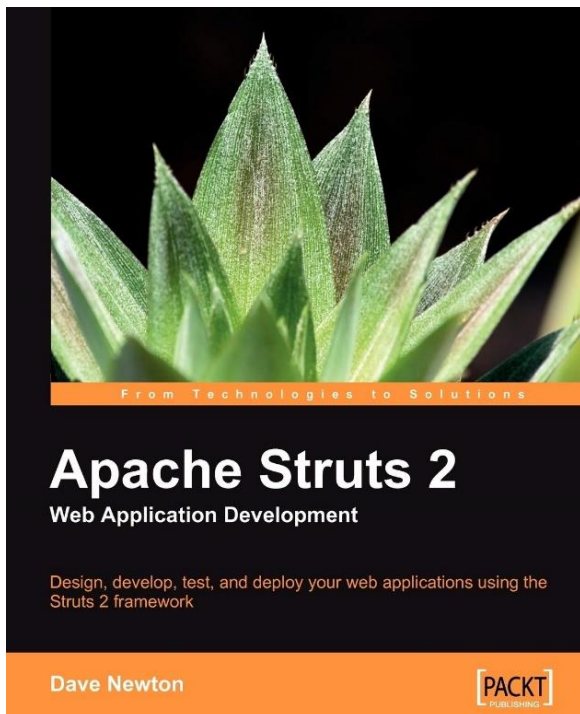
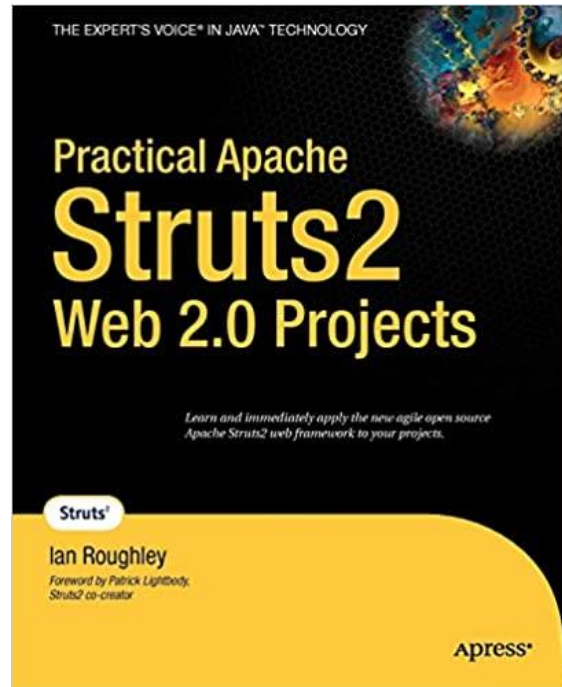
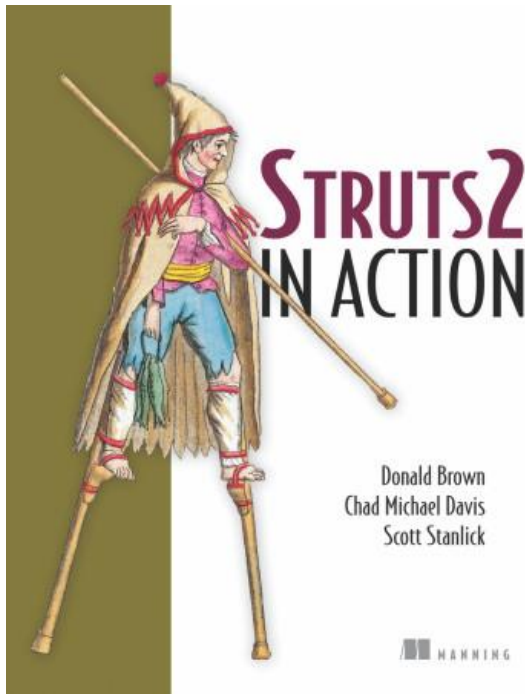
```
<!-- ===== -->
<%@ include file="../includes/notific.jsp"%>
<!-- ===== -->
```

## h) Packet Management

Chia nhỏ công việc ra thành các module con, theo chức năng, nhiệm vụ, loại, mỗi module thực hiện một công việc khi nào cần thì gọi module đó, giúp quản lý dễ dàng hơn, mã nguồn sẽ clean thêm một chút.



## 9. Tài liệu tham khảo



- Source Code & Documents: <https://github.com/volam1999/Share-Rooms>
- NICE ADMIN: <https://github.com/hacktheme/Nice-Admin>