

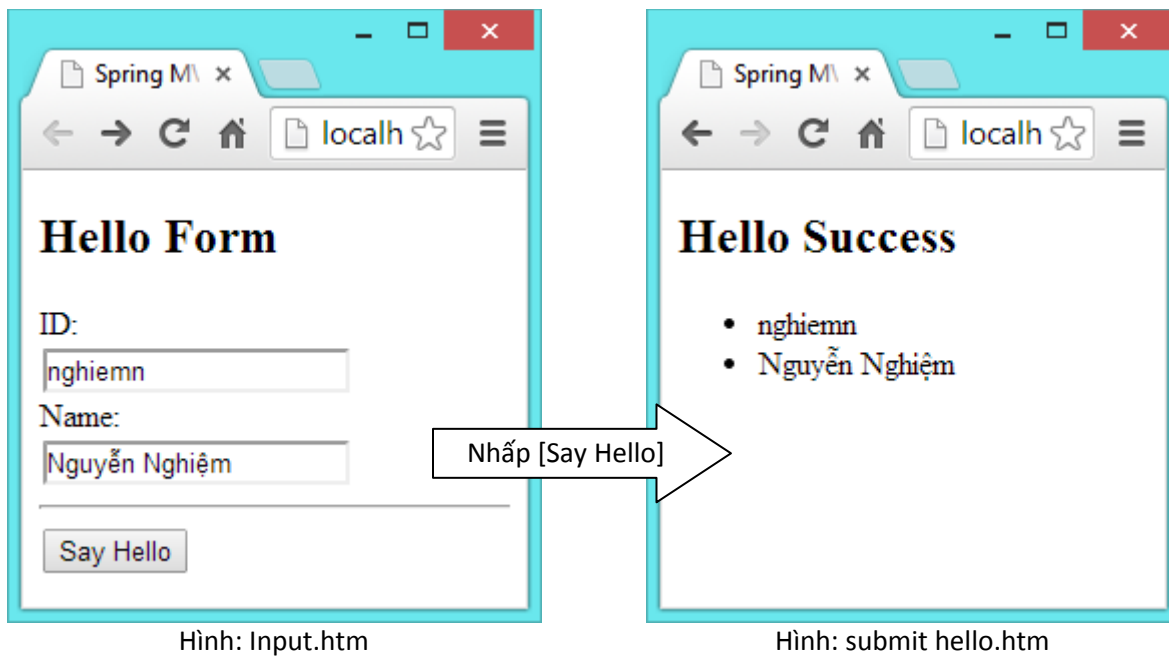
MỤC TIÊU

Kết thúc bài thực hành, bạn có khả năng

Tạo một project Spring MVC với eclipse

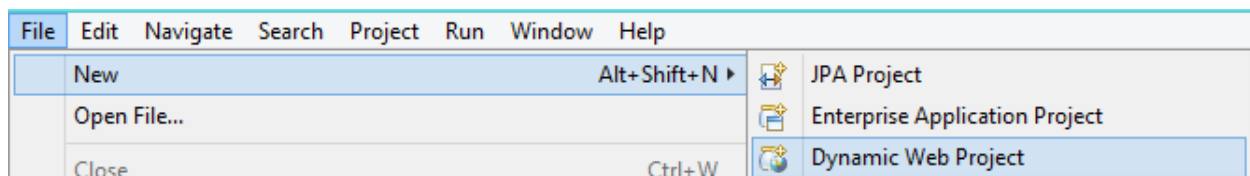
- ✓ Viết một Controller
- ✓ Tạo một View
- ✓ Cách chia sẻ dữ liệu giữa Controller và View thông qua model

MÔ TẢ

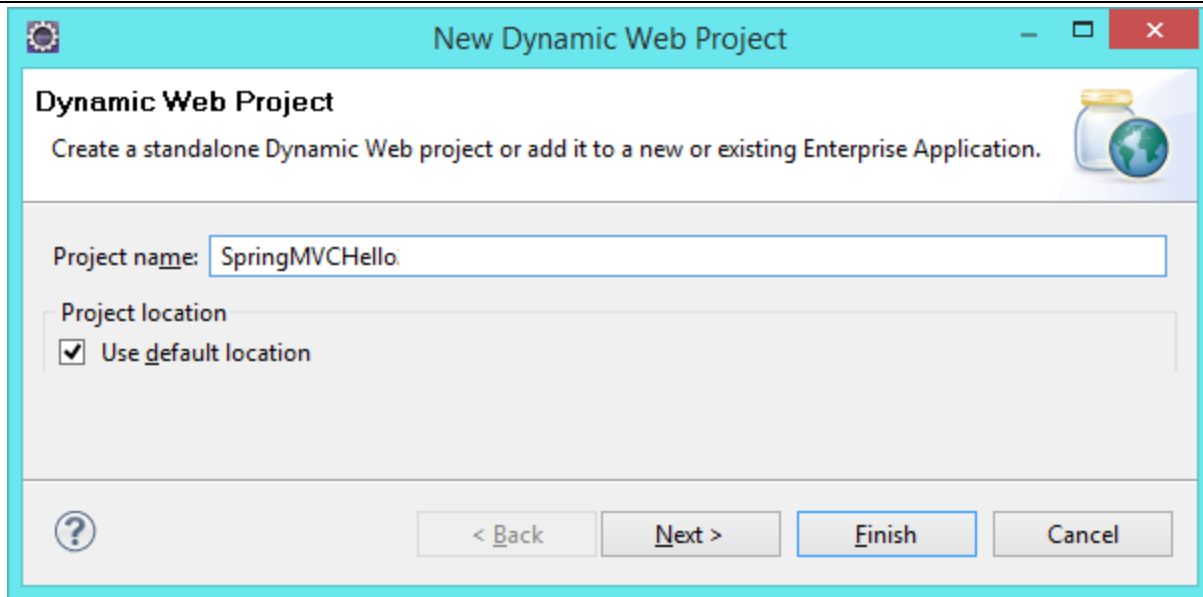


THỰC HIỆN

Bước 1: Tạo dự án



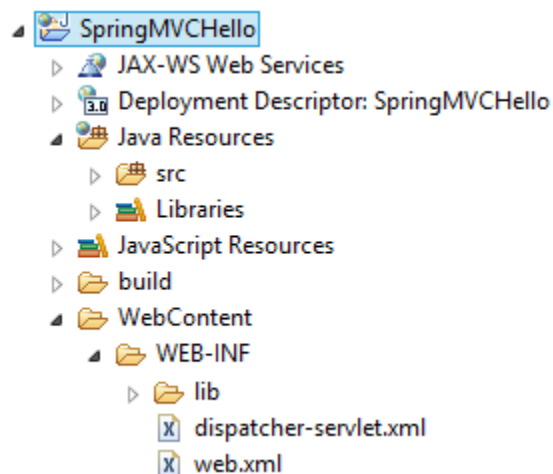
Hình: Tạo dự án



Hình: Đặt tên dự án

Bước 2: Cấu hình dự án

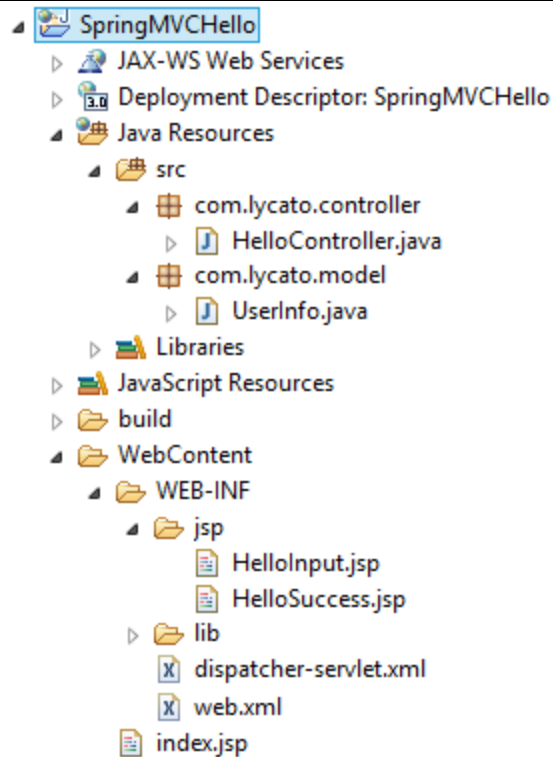
Chép web.xml, dispatcher-servlet.xml, index.jsp và lib/* .jar từ SpringMVCBlank vào vị trí tương ứng của SpringMVCHello



Hình: Cấu trúc dự án sau khi bổ sung từ SpringMVCBlank

Bước 3: Bổ sung các thành phần

Thêm các lớp HelloController, UserInfo và 2 file jsp HelloInput.jsp và HelloSuccess.jsp vào các vị trí như hình sau



Hình: Cấu trúc đầy đủ của dự án

Bước 4: Viết mã

Soạn nội dung cho các thành phần vừa thêm vào như sau:

HelloController.java

```
@Controller
public class HelloController {

    @RequestMapping(value="input")
    public String showForm(ModelMap model) {
        model.addAttribute("user", new UserInfo());
        return "HelloInput";
    }

    @RequestMapping(value="hello")
    public String sayHello(@ModelAttribute("user") UserInfo user) {
        return "HelloSuccess";
    }
}
```

Các thành phần quan trọng

- @Controller: chỉ ra class này là controller
- @RequestMapping(value="action") định nghĩa một action
- model.addAttribute(name, value): thêm một attribute vào model
- @ModelAttribute("user"): thêm đối tượng user vào model
- return ViewName: chọn view để hiển thị

UserInfo.java

Lớp sau được sử dụng để chứa thông tin user được nhập vào form.

```
public class UserInfo {
    String id;
    String name;

    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

HelloInput.jsp

```
<%@page pageEncoding="utf-8" contentType="text/html; charset=utf-8" %>
<%@taglib uri="http://www.springframework.org/tags/form" prefix="form"%>

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Spring MVC Seminar 2014</title>
</head>
<body>
    <h2>Hello Form</h2>
    <form:form action="hello.htm"
        method="post" modelAttribute="user">
        <div>ID:</div>
        <form:input path="id"/>

        <div>Name:</div>
        <form:input path="name"/>
        <hr>
        <input type="submit" value="Say Hello">
    </form:form>
</body>
</html>
```

- `<%@taglib%>` khai báo thư viện thẻ xây dựng form
- `<form:form>` tạo form. Thuộc tính `modelAttribute` chỉ ra bean được sử dụng để hiển thị lên form
- `<form:input>` tạo ô nhập. Thuộc tính `path="property"` tên thuộc tính của bean `UserInfo`

HelloSuccess.jsp

```
<%@page pageEncoding="utf-8" contentType="text/html; charset=utf-8" %>

<!DOCTYPE html>
```

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Spring MVC Seminar 2014</title>
</head>
<body>
  <h2>Hello Success</h2>
  <ul>
    <li>${user.id}</li>
    <li>${user.name}</li>
  </ul>
</body>
</html>
```

- \${user.id}: hiển thị thuộc tính id của bean user đặt trong model

Index.jsp

```
<jsp:forward page="input.htm"/>
```

Bước 4: Nâng cấp

- Thêm ô nhập email vào form
- Hiển thị email nhập vào lên trang kết quả