

## MỤC TIÊU:

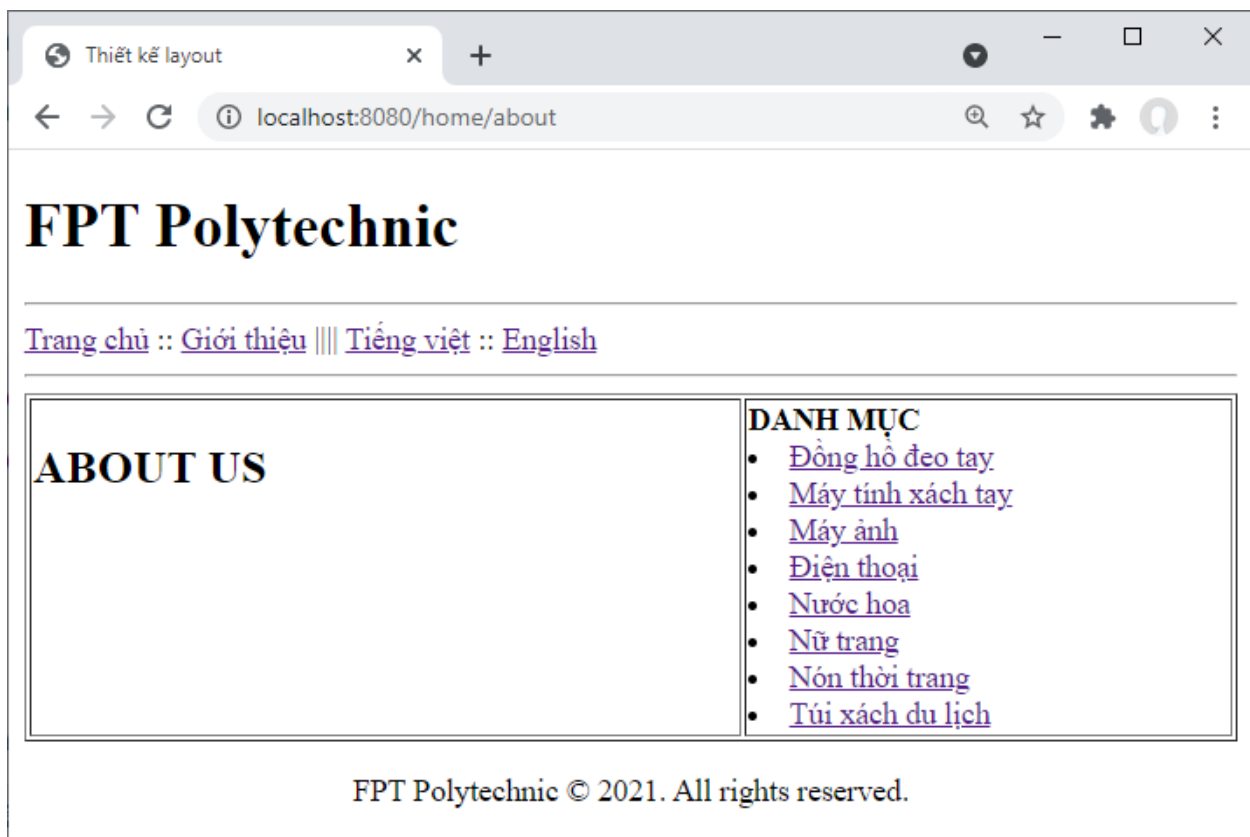
Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng Tiles Framework để xây dựng layout
- ✓ Tổ chức website đa ngôn ngữ

## PHẦN I

### Bài 1 (2 điểm)

Sử dụng Tiles Framework để tổ chức layout cho ứng dụng web như hình sau



Trong đó các vùng được đặt tên và tổ chức như sau

_header.jsp	
_menu.jsp	
	_aside.jsp
_footer.jsp	

Hãy thực hiện yêu cầu trên theo hướng dẫn sau

### 1. Khai báo thư viện cần thiết

```
<dependency>
  <groupId>org.apache.tiles</groupId>
  <artifactId>tiles-jsp</artifactId>
  <version>3.0.8</version>
</dependency>
```

### 2. Cấu hình ViewResolver và TilesConfigurer

```
@Configuration
public class TilesConfig {
    @Bean("viewResolver")
    public ViewResolver getViewResolver() {
        UrlBasedViewResolver resolver = new UrlBasedViewResolver();
        resolver.setViewClass(TilesView.class);
        return resolver;
    }
    @Bean("tilesConfigurer")
    public TilesConfigurer getTilesConfigurer() {
        TilesConfigurer tilesConfigurer = new TilesConfigurer();
        tilesConfigurer.setDefinitions("/views/layout/tiles.xml");
        return tilesConfigurer;
    }
}
```

### 3. Tổ chức tiles.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE tiles-definitions PUBLIC
    "-//Apache Software Foundation//DTD Tiles Configuration 3.0//EN"
    "http://tiles.apache.org/dtds/tiles-config_3_0.dtd">
<tiles-definitions>
  <definition name="layout" template="/views/layout/index.jsp">
    <put-attribute name="header" value="/views/layout/_header.jsp" />
    <put-attribute name="menu" value="/views/layout/_menu.jsp" />
    <put-attribute name="body" value="" />
    <put-attribute name="aside" value="/views/layout/_aside.jsp" />
  </definition>
</tiles-definitions>
```

```
<put-attribute name="footer" value="/views/layout/_footer.jsp" />
</definition>
<definition name="*/*" extends="layout">
    <put-attribute name="body" value="/views/{1}/{2}.jsp" />
</definition>
</tiles-definitions>
```

#### 4. Thiết kế giao diện layout

- index.jsp

```
<%@ page pageEncoding="utf-8"%>
<%@ taglib uri="http://tiles.apache.org/tags-tiles" prefix="tiles" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Thiết kế layout</title>
</head>
<body>
    <tiles:insertAttribute name="header"/>
    <tiles:insertAttribute name="menu"/>
    <table style="width:100%" border="1">
        <tr>
            <td valign="top"><tiles:insertAttribute name="body"/></td>
            <td width="250"><tiles:insertAttribute name="aside"/></td>
        </tr>
    </table>
    <tiles:insertAttribute name="footer"/>
</body>
</html>
```

- \_header.jsp

```
<%@ page pageEncoding="utf-8"%>
<header>
    <h1>FPT Polytechnic</h1>
</header>
```

- \_menu.jsp

```
<%@ page pageEncoding="utf-8"%>
```

```
<nav>
    <hr>
    <a href="/home/index">Trang chủ</a> ::
    <a href="/home/about">Giới thiệu</a> |||
    <a href="?lang=vi">Tiếng việt</a> ::
    <a href="?lang=en">English</a>
    <hr>
</nav>
```

- \_aside.jsp

```
<%@ page pageEncoding="utf-8"%>
<b>DANH MỤC</b>
<ul>
    <li><a href="">Đồng hồ đeo tay</a></li>
    <li><a href="">Máy tính xách tay</a></li>
    <li><a href="">Máy ảnh</a></li>
    <li><a href="">Điện thoại</a></li>
    <li><a href="">Nước hoa</a></li>
    <li><a href="">Nữ trang</a></li>
    <li><a href="">Nón thời trang</a></li>
    <li><a href="">Túi xách du lịch</a></li>
</ul>
```

- \_footer.jsp

```
<%@ page pageEncoding="utf-8"%>
<footer style="text-align: center;">
    <p>FPT Polytechnic &copy; 2021. All rights reserved.</p>
</footer>
```

## 5. HomeController

```
@Controller
public class HomeController {
    @RequestMapping("/home/index")
    public String index() {
        return "home/index";
    }
    @RequestMapping("/home/about")
    public String about() {
```

```
        return "home/about";  
    }  
}
```

View index.jsp

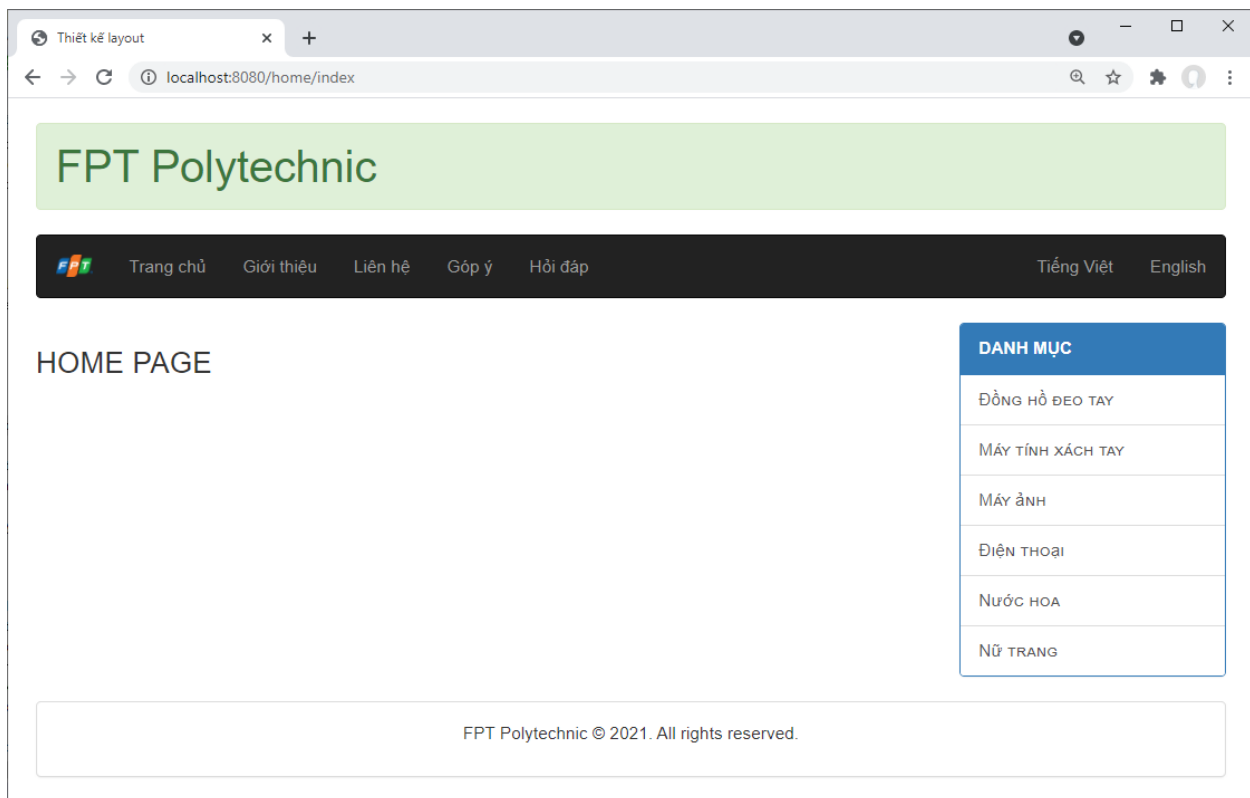
```
<%@ page pageEncoding="utf-8"%>  
<h3>HOME PAGE</h3>
```

View about.jsp

```
<%@ page pageEncoding="utf-8"%>  
<h2>ABOUT US</h2>
```

## Bài 2 (2 điểm)

Hãy nâng cấp layout của bài 1 để có được giao diện như sau



Hướng dẫn: Sử dụng bootstrap

## Bài 3 (1 điểm)

Giảng viên cho thêm

## PHẦN II

### Bài 4 (2 điểm)

Hãy làm website đa ngôn ngữ cho menu trong layout/index.jsp và view home/index.jsp của bài 1 để khi nhấp chuột vào liên kết của ngôn ngữ thì giao diện được thay đổi theo ngôn ngữ được chọn.

Hãy thực hiện theo các bước sau đây

#### 1. Tổ chức tài nguyên đa ngôn ngữ

Trong thư mục **src/main/resources/i18n** tạo các file tài nguyên đa ngôn ngữ có tên và nội dung như sau

<b>layout.properties</b>	<b>layout_vi.properties</b>
lo.mn.home=Home lo.mn.about=About Us	lo.mn.home=Trang chủ lo.mn.about=Giới thiệu
<b>home.properties</b>	<b>home_vi.properties</b>
home.title=HOME PAGE	home.title=TRANG CHỦ

#### 2. Cấu hình nạp tài nguyên vào hệ thống

```
@Configuration
public class I18NConfig implements WebMvcConfigurer{
    @Bean("messageSource")
    public MessageSource getMessageSource() {
        ReloadableResourceBundleMessageSource ms
            = new ReloadableResourceBundleMessageSource();
        ms.setBasenames("classpath:i18n/layout", "classpath:i18n/home");
        ms.setDefaultEncoding("utf-8");
        return ms;
    }
}
```

#### 3. Hiển thị tài nguyên đa ngôn ngữ

- Layout/index.jsp

```
<%@ page pageEncoding="utf-8"%>
<%@ taglib uri="http://www.springframework.org/tags" prefix="s" %>
<nav>
```

```
<hr>
<a href="/home/index"><s:message code="lo.mn.home"/></a> ::
<a href="/home/about"><s:message code="lo.mn.about"/></a> |||
<a href="?lang=vi">Tiếng việt</a> ::
<a href="?lang=en">English</a>
<hr>
</nav>
```

- Home/index.jsp

```
<%@ page pageEncoding="utf-8"%>
<%@ taglib uri="http://www.springframework.org/tags" prefix="s" %>
<h3><s:message code="home.title"/></h3>
```

4. Cấu hình *tham số thay đổi ngôn ngữ* và *cơ chế duy trì* ngôn ngữ được chọn bằng cách bổ sung vào file cấu hình ở trên đoạn mã sau

```
@Override
public void addInterceptors(InterceptorRegistry registry) {
    LocaleChangeInterceptor locale = new LocaleChangeInterceptor();
    locale.setParamName("lang");
    registry.addInterceptor(locale).addPathPatterns("/*");
}

@Bean("localeResolver")
public LocaleResolver getLocaleResolver() {
    CookieLocaleResolver resolver = new CookieLocaleResolver();
    resolver.setDefaultLocale(new Locale("vi"));
    resolver.setCookieMaxAge(10*24*60*60); // 10 ngày
    resolver.setCookiePath("/");
    return resolver;
}
```

5. Thay đổi ngôn ngữ

Request đến một URL bất kỳ với tham số *lang*, LocaleChangeInterceptor sẽ thực hiện công việc thay đổi ngôn ngữ theo cookie trên máy của bạn.

```
<a href="?lang=vi">Tiếng việt</a> ::
<a href="?lang=en">English</a>
```

Giải pháp trên đây thực hiện được với các tình huống đơn giản vì có thể thiếu tham số của trang hiện tại hoặc hành động trước đó là POST.

Để có giải pháp toàn vẹn hơn, bạn nên sử dụng ajax để gọi đến một url với tham số lang để LocaleChangeInterceptor thay đổi ngôn ngữ sau đó gọi reload trang web hiện tại.

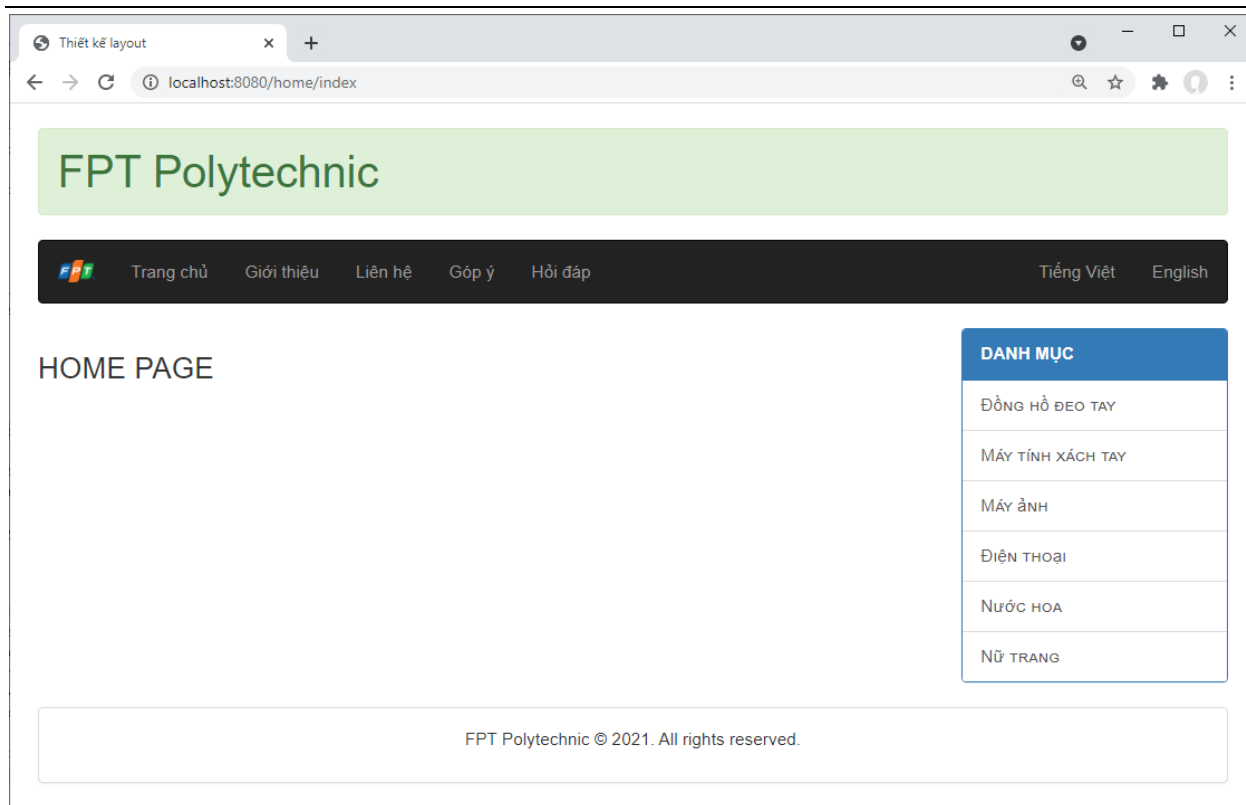
Để thực hiện điều này cần những thư viện jQuery và viết đoạn mã jQuery và \_menu.jsp:

```
<script>
$(document).ready(function(){
    $("a[href*=lang]").on("click", function(){
        var param = $(this).attr("href");
        $.ajax({
            url: "/home/index" + param,
            success: function(){
                location.reload();
            }
        });
        return false;
    })
})
</script>
```

### Bài 5 (2 điểm)

Hãy tổ chức đa ngôn ngữ cho layout của riêng bạn trong bài 2 và các view khác như about.jsp và \_aside.jsp, \_footer.jsp.





## Bài 6 (1 điểm)

Giảng viên cho thêm