

FAQ: How to Add a Public Landing Page

Question

I want my application to have a landing page that is accessible to everyone at first, the user can then signup or login to access specific features. Our current framework only points to the login page. How do i implement this scenario?

Answer

Good question!

You can add a public view page and set up the security constraints to allow access to the view page.

In this example, we have a view page that anyone can access. Then they can click the link to access the secure pages.



Source Code

Here is a working code example of the project: [spring-security-landing-page.zip](#)

This project has the following mods

1. Updated security configs to allow public access to landing page
2. Updated controller to send requests to landing page
3. New landing page

Details below

1. Updated security configs to allow public access to landing page

See the config below. It will "permit all" access to the landing page "/". Also, for successful logout it will redirect to the landing page "/"

File: DemoSecurityConfig.java

```
1.  @Override
2.  protected void configure(HttpSecurity http) throws Exception {
3.      http.authorizeRequests()
4.          .antMatchers("/").permitAll() // allow public access to home page
5.          .antMatchers("/employees").hasRole("EMPLOYEE")
6.          .antMatchers("/leaders/**").hasRole("MANAGER")
7.          .antMatchers("/systems/**").hasRole("ADMIN")
8.          .and()
9.          .formLogin()
10.         .loginPage("/showMyLoginPage")
11.         .loginProcessingUrl("/authenticateTheUser")
12.         .permitAll()
13.         .and()
14.         .logout()
15.         .logoutSuccessUrl("/") // after logout then redirect to landing page (root
16.     )
17.     .permitAll();
18. }
```

2. Updated controller to send requests to landing page

In the controller file, added new "/" mapping to send to landing page. And changed the original home mapping to "/employees". see changes in bold.

File: DemoController.java

```
1.     @GetMapping("/")
2.     public String showLanding() {
3.         return "landing";
4.     }
5.
6.     @GetMapping("/employees")
7.     public String showHome() {
8.         return "home";
9.     }
```

3. New landing page

Created a new view page for landing information. Anyone can access this page

File: src/main/webapp/WEB-INF/view/landing.jsp

```
1. <html>
2.     <head>
3.         <title>luv2code Landing Page</title>
4.     </head>
5.     <body>
6.         <h2>luv2code Landing Page</h2>
7.         <hr>
8.         <p>
9.             Welcome to the landing page! This page is open to the public ... no login required :-)
10.        </p>
11.        <hr>
12.        <p>
13.            <a href="${pageContext.request.contextPath}/employees">Access Secure Site (requires login)</a>
14.        </p>
15.    </body>
16. </html>
```

To test the application, be sure to open a new private/incognito window.

Now when you run the application, anyone can access the public home page without logging in. If they follow through on the link, then they will be required to login.

Resources for this lecture

- [spring-security-landing-page.zip](#)