

Securyty App Example

**1.** <?xml version="1.0" encoding="UTF-8"?>

**2.** <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

**3.** xmlns="http://java.sun.com/xml/ns/javaee"

**4.** xmlns:web="http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"

**5.** xsi:schemaLocation="http://java.sun.com/xml/ns/javaee

**6.** http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"

**7.** version="2.5">

**8.** <servlet>

**9.** <servlet-name>Faces Servlet</servlet-name>

**10.** <servlet-class>javax.faces.webapp.FacesServlet</servlet-class>

**11.** </servlet>

**12.** <servlet-mapping>

**13.** <servlet-name>Faces Servlet</servlet-name>

**14.** <url-pattern>/faces/\*</url-pattern>

**15.** </servlet-mapping>

**16.** <welcome-file-list>

**17.** <welcome-file>index.xhtml</welcome-file>

**18.** </welcome-file-list>

**19.** <context-param>

**20.** <param-name>javax.faces.PROJECT\_STAGE</param-name>

**21.** <param-value>Development</param-value>

**22.** </context-param>

**23.**

**24.** <security-constraint>

**25.** <web-resource-collection>

**26.** <web-resource-name>Protected Pages</web-resource-name>

**27.** <url-pattern>/faces/protected/\*</url-pattern>

**28.** <url-pattern>/protected/\*</url-pattern>

**29.** </web-resource-collection>

**30.** <auth-constraint>

**31.** <role-name>registereduser</role-name>

**32.** <role-name>invitedguest</role-name>

**33.** </auth-constraint>

**34.** </security-constraint>

**35.** <security-role>

**46.** <role-name>registereduser</role-name>

**47.** </security-role>

**48.** <security-role>

**49.** <role-name>invitedguest</role-name>

**50.** </security-role>

**51.** </web-app>

**36.** <login-config>

**37.** <auth-method>FORM</auth-method>

**38.** <realm-name>corejsfRealm</realm-name>

**39.** <form-login-config>

**40.** <form-login-page>/login.html</form-login-page>

**41.** <form-error-page>/noauth.html</form-error-page>

**42.** </form-login-config>

**43.** </login-config>

accesscontrol/web/noauth.html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"

**2.** "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

**3.** <html xmlns="http://www.w3.org/1999/xhtml">

**4.** <head>

**5.** <title>Authentication failed</title>

**6.** </head>

**7.**

**8.** <body>

**9.** <p>Sorry--authentication failed. Please try again.</p>

**10.** </body>

**11.** </html>

package com.corejsf;

**2.**

**3.** import java.io.Serializable;

**4.** import java.util.logging.Logger;

**5.**

**6.** import javax.inject.Named;

**7.** // or import javax.faces.bean.ManagedBean;

**8.** import javax.enterprise.context.SessionScoped;

**9.** // or import javax.faces.bean.SessionScoped;

**10.** import javax.faces.context.ExternalContext;

**11.** import javax.faces.context.FacesContext;

**12.** import javax.servlet.http.HttpServletRequest;

**13.**

**14.** @Named("user") // or @ManagedBean(name="user")

**15.** @SessionScoped

**16.** public class UserBean implements Serializable {

**17.** private String name;

**18.** private String role;

**19.** private static Logger logger = Logger.getLogger("com.corejsf");

**20.**

**21.** public String getName() {

**22.** if (name == null) getUserData();

return name == null ? "" : name;

**24.** }

**25.**

**26.** public String getRole() { return role == null ? "" : role; }

**27.** public void setRole(String newValue) { role = newValue; }

**28.**

**29.** public boolean isInRole() {

**30.** ExternalContext context

**31.** = FacesContext.getCurrentInstance().getExternalContext();

**32.** Object requestObject = context.getRequest();

**33.** if (!(requestObject instanceof HttpServletRequest)) {

**34.** logger.severe("request object has type " + requestObject.getClass());

**35.** return false;

**36.** }

**37.** HttpServletRequest request = (HttpServletRequest) requestObject;

**38.** return request.isUserInRole(role);

**39.** }

**40.**

**41.** private void getUserData() {

**42.** ExternalContext context

**43.** = FacesContext.getCurrentInstance().getExternalContext();

**44.** Object requestObject = context.getRequest();

**45.** if (!(requestObject instanceof HttpServletRequest)) {

**46.** logger.severe("request object has type " + requestObject.getClass());

**47.** return;

**48.** }

**49.** HttpServletRequest request = (HttpServletRequest) requestObject;

**50.** name = request.getRemoteUser();

**51.** }

**52.** }

accesscontrol/src/java/com/corejsf/messages.properties

title=Authentication successful

**2.** youHaveAccess=You now have access to protected information!

**3.** yourUserName=Your user name

**4.** memberOf=Member of