EH203 - Week 3 - Project

# Specification

You are to develop a basic login system for a website. The website only needs to include a couple of test pages, to verify that the login system works.

The web app will read the user accounts from a file, which can use any suitable format, though you may find the format below useful:

username1|password1|name1

username2|password2|name2

You provided with a very basic example of such a web app, which is accessible at:

http://intare:8080/eh203-login

# Additional Requirements

1. Use JSP includes for the site header and footer. Page title, contained in the header JSP, should be customizable using a parameter
2. Store something in the session to determine if a user has been authenticated
3. If request is made for a page other than the login page, and a user has not been authenticated, then *redirect* to the login page. To redirect a response to another URL, we use the sendRedirect() method of the HttpServletResponse object, e.g.  
    response.sendRedirect(“login.htm”);
4. When user successfully authenticates, then redirect back to the *original* page, or the home page if there was no original page
5. Create a TAG file which displays the name of the currently logged in user

# Extra Credit

1. Create a *custom tag* which can be placed at the top of every JSP page to cause a redirect to the login page, when no user has been authenticated. This tag should have two attributes

* loginURL: the URL of the login page
* redirectURL: the current URL, which the login page should redirect back to

For example, on index.jsp this tag could be used as follows:

<ehsdi:authCheck loginURL="login.htm" redirectURL="index.jsp" />

## Useful Snippets

### Accessing the request object in a custom tag

The SimpleTagSupport class has a getJspContext() method which returns the JspContext object - which can then be cast to a PageContext object. It in turn has a getRequest() method which returns a ServletRequest object, which can be cast to a HttpServletRequest object, which can then be used to access things like sessions

PageContext context = (PageContext)getJspContext();  
HttpServletRequest request = (HttpServletRequest)context.getRequest();