

CHAPTER 4

Bullhorn Site Overview

The diagram in Figure 4-1 illustrates how the site fits together. Only the core components are shown. You may wish to add additional pages and classes as needed.

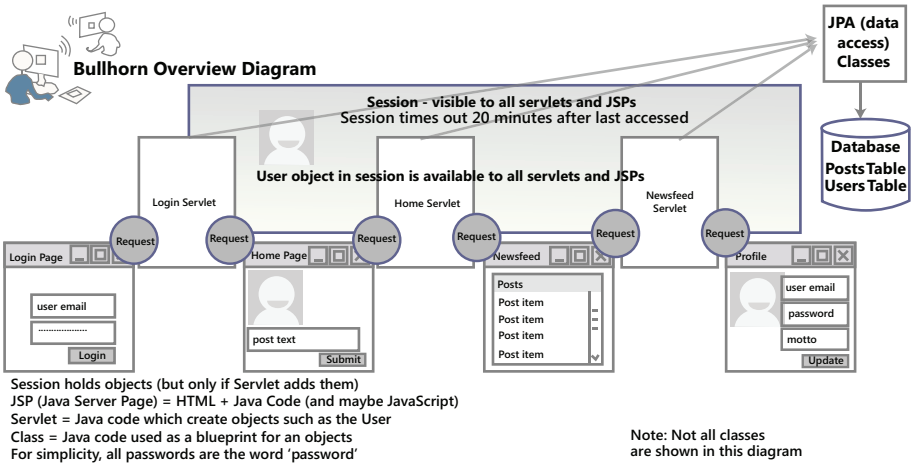


Figure 4-1. The components that make up Bullhorn

The Components of Bullhorn

- **Servlets** Java classes that extend the web server to provide an interface to the browser and database or other servlets.

- **JPA classes** Java Persistence API classes that are used to communicate between the servlets and the database.
- **Request objects** represent information sent between the browser and the servlets. This information might include email addresses and passwords that are being used by the servlet to allow access to the site.
- **Sessions** are the web server's method or approach of retaining data while the user is accessing the site.
- **User objects** The user information is stored in a class that will be stored in the session and is available to all pages for the current user.
- **JSP** (Java Server Pages) Web pages that contain HTML and tags from the JSP Standard Tag Library to add functionality. Because they contain code they can dynamically render for each user's request. The JSP Standard Tag Library permits each person to view their own version of the page.
- **HTML** (Hypertext Markup Language) pages. HTML is a system for tagging text files to control fonts, colors, and images on your web pages.

Tip To keep your HTML from getting too complicated, use CSS (Cascading Style Sheets) and JavaScript to control the presentation of your content and let HTML control the layout.

The Bullhorn application contains web pages for login, home, news feed, and user profile. The user starts at the login page. Once the user clicks the Login button, the request (data from the login form) will be sent to the login servlet.

The login servlet will validate the user against the database. A valid user will be stored in the session, which is the website's way of remembering data between page views. Invalid users will not get past the login page until they enter a correct username and password combination.

We will create other objects (classes) to validate data or support the classes and pages shown in the diagram.

What Does Each Page Look Like?

The login page will contain text boxes in which the user will enter their email and password. This information will be verified in the login servlet. If they match what is in the database then the user will be redirected to their home page. If they do not match then the user will be prompted to log in again. Users who reach the login page but aren't registered on the site can register for a login by clicking on the "Join" link. See Figure 4-2.

Login



The screenshot shows a web page titled "Login". Below the title is the heading "Please sign in". There are two text input fields: "Email address" and "Password". Below these fields is a large blue button labeled "Sign in". To the left of the "Sign in" button is a link labeled "Join". At the bottom of the page, there is a footer with the text "Bullhorn Web Site ©2017".

Figure 4-2. The login page contains text boxes for email and password and a button to sign in to the application

The home page will allow each user to create a new post. Each post is limited to 141 characters, so the home page enforces this restriction (see Figure 4-3). Once the user is logged in, all pages contain a navigation bar at the top that allows the user to navigate to different pages, view or edit their profile, and search for posts containing a specific word.

The screenshot shows the Bullhorn website's home page. At the top is a blue navigation bar containing a megaphone logo, the word "Bullhorn", and several menu items: "Home", "News Feed", "User Options" (with a dropdown arrow), a search box with the text "Search", and a "Submit" button. Below the navigation bar, the main heading reads "This is the home page". Underneath is a section titled "Create New Post (141 char):" which contains a text input field with the placeholder text "Express yourself!". Below the input field, it says "141 characters remaining". At the bottom of this section are two buttons: "Submit" and "Clear". At the very bottom of the page is a light gray footer containing a speaker icon, the text "Bullhorn Web Site", and a copyright symbol followed by "©2017".

Figure 4-3. The home page contains a form to submit a post to the database. The form contains a text box and buttons to either submit the post or clear the form.

Each page contains the same navigation bar, which allows the user to move around the application. The navigation bar contains the logo, links for the home page and the news feed page, and a search box. It also displays the name of the logged-in user. The user can also select from various user options, which is implemented as a drop-down list. These include logging out, viewing or editing profile, and submitting feedback. See Figure 4-4.



Figure 4-4. The navigation bar in Bullhorn shows at the top of every page

The “News Feed” link in the navigation bar will take a user to the news feed page, which displays all posts from all users. Each user’s email address is a link that will display the user’s profile information. Clicking Search in the navigation bar will also display the news feed, but filtered to posts that contain the text entered in the search text box. See Figure 4-5.

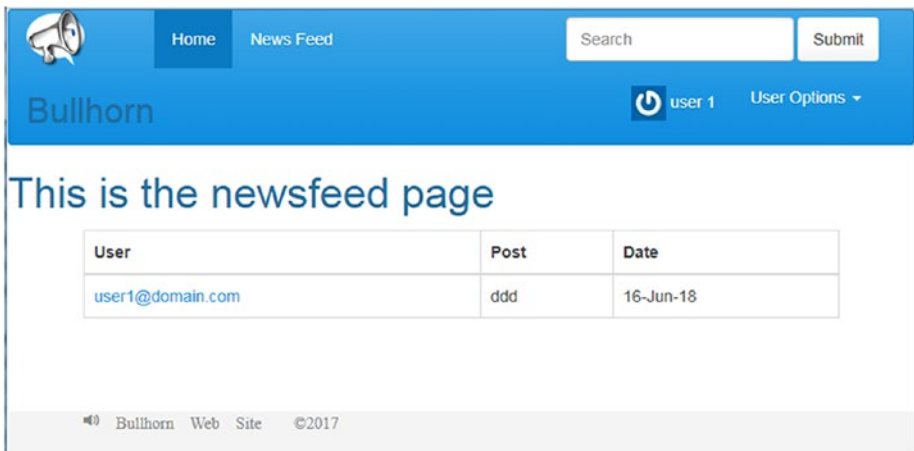


Figure 4-5. The news feed page displays any posts that are in the database

The profile for a user is read-only. It displays their email, motto, join date, and avatar image, if any. Users can view profiles for other users by clicking on their user names from the news feed page. See Figure 4-6.

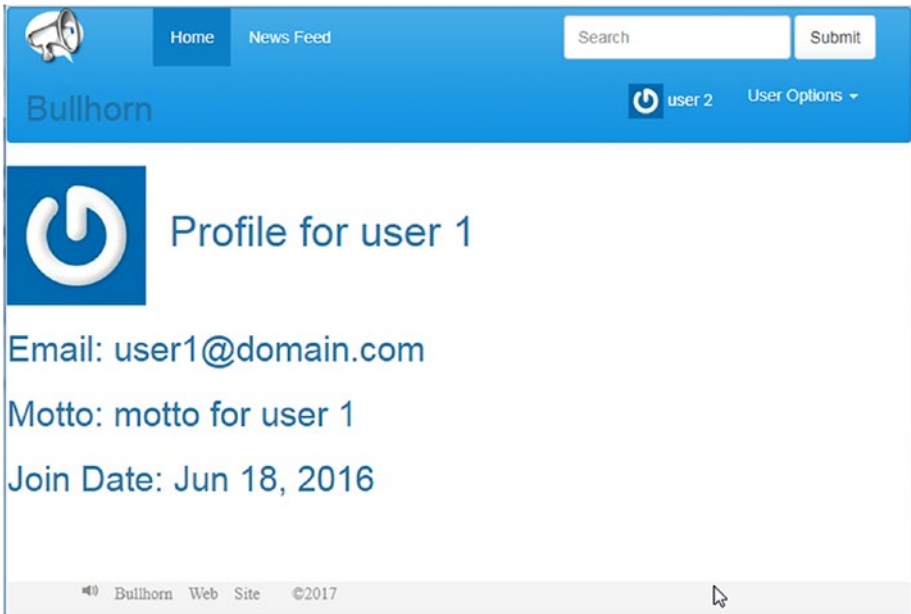


Figure 4-6. The profile page for a user shown in read-only view

Editing a Profile

If a user views their own profile, then the profile can be edited. See Figure 4-7.

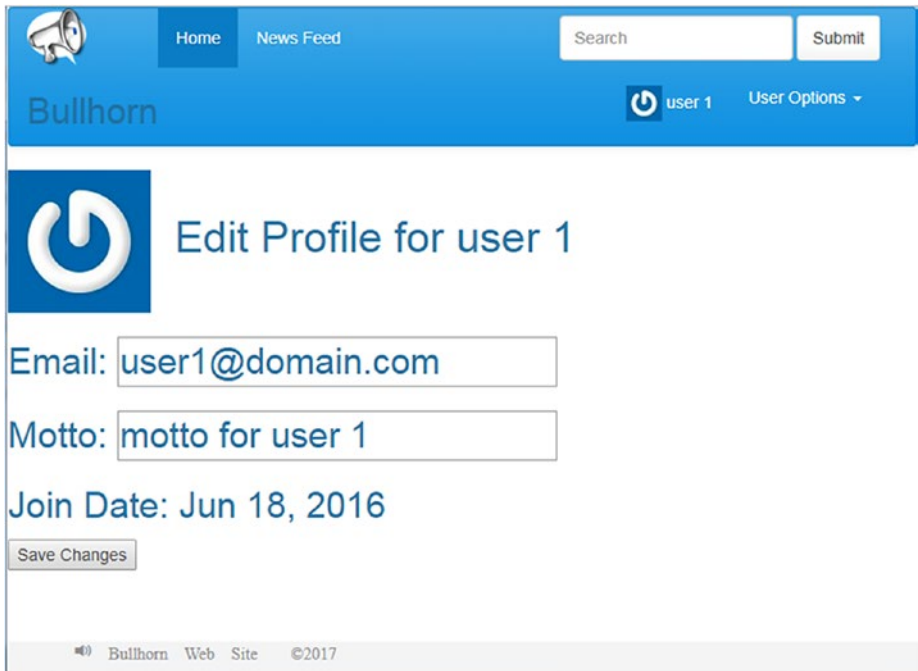


Figure 4-7. The profile page for the logged-in user displays with text boxes and a button so the user can make changes

The support page doesn't show much, just some text to let you know it exists. We could modify this to include a text box that will send an email or add a record to the database. Then, the support person could check for new messages periodically. See Figure 4-8.

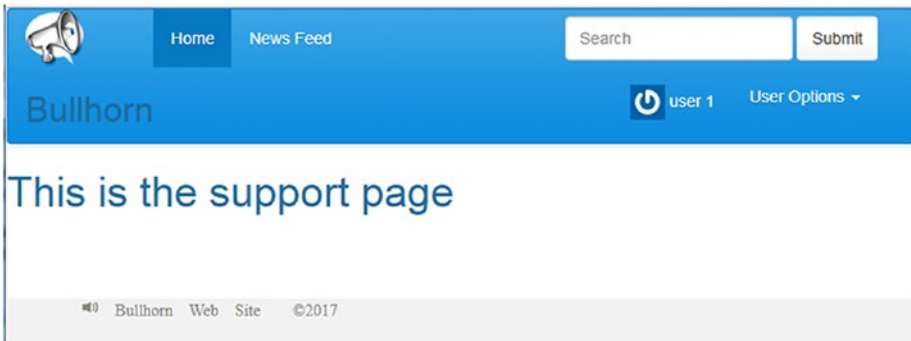


Figure 4-8. *The support page could allow you to let users submit requests to the web administrator*