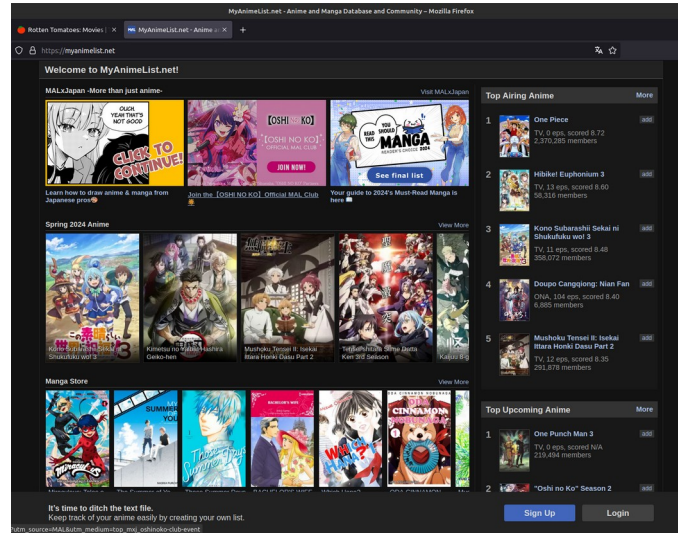
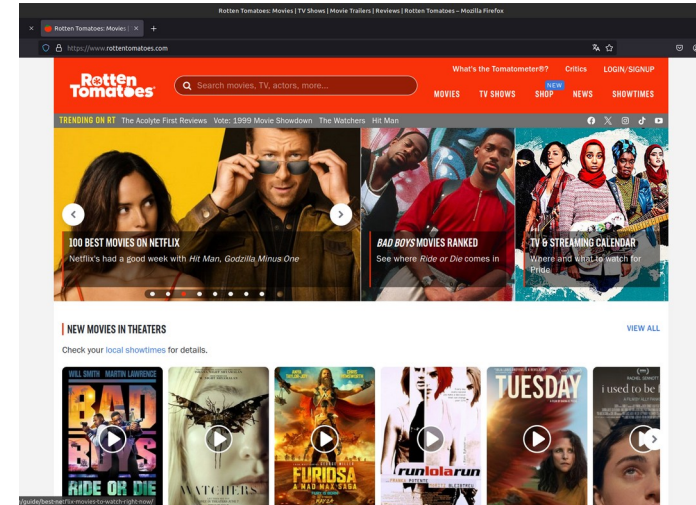


Inspiration

The idea came from 2 major websites for every movies enjoyer, cinema critics, anime watcher and manga reader...



MyAnimeList



Rotten Tomatoes

Our web application does not have has many functionalities or such a responsive design, *but* it is built with those advance features in mind

BookRec

What is BookRec?

BookRec is a web application based on Java Servlet, which lets users register to the website and search for books, authors and other users

How was it built?

It was built using mostly Java technology (Java Servlets and DAOs), HTML (dynamic content provided by JSP pages), CSS and some JavaScript.



BookRec

What about the database?

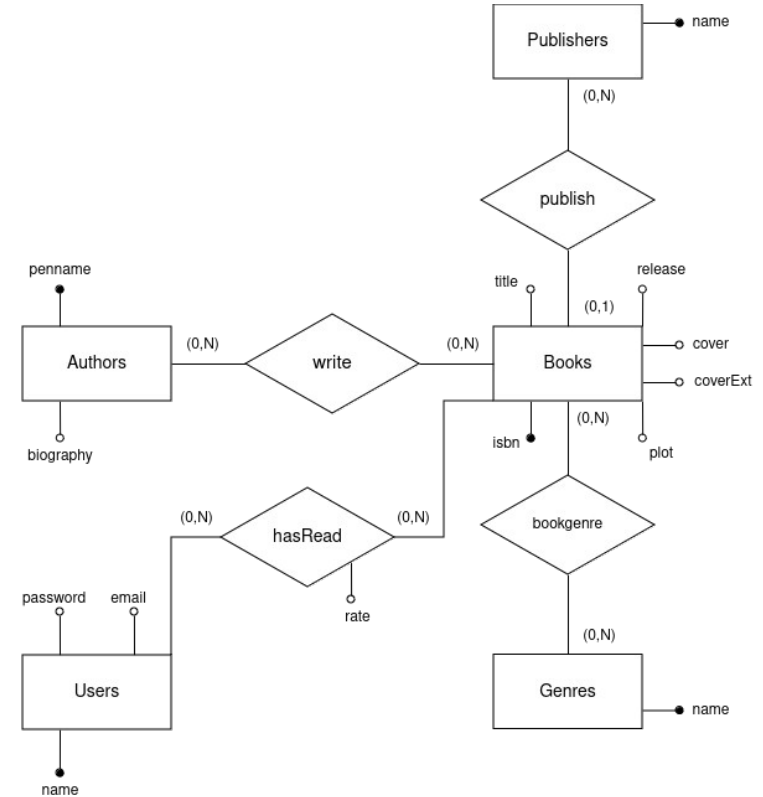
The database used for this project is a simple relational database implemented with PostgreSQL.



It's based on 3 main entities:

- Books
- Authors
- Users

Around this entities, other tables are built to add extra functionalities (what publisher has released the book, which are the authors of this book, what genres a book is, ...)

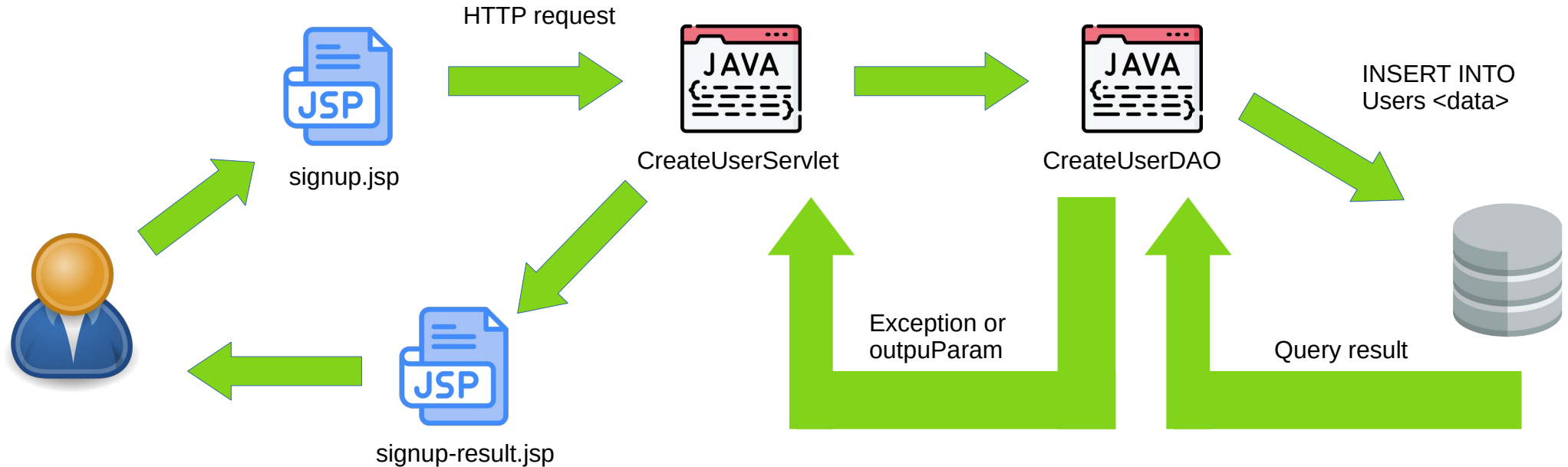


ER schema of the actual database

**Following some example of the
workflow of the web application**

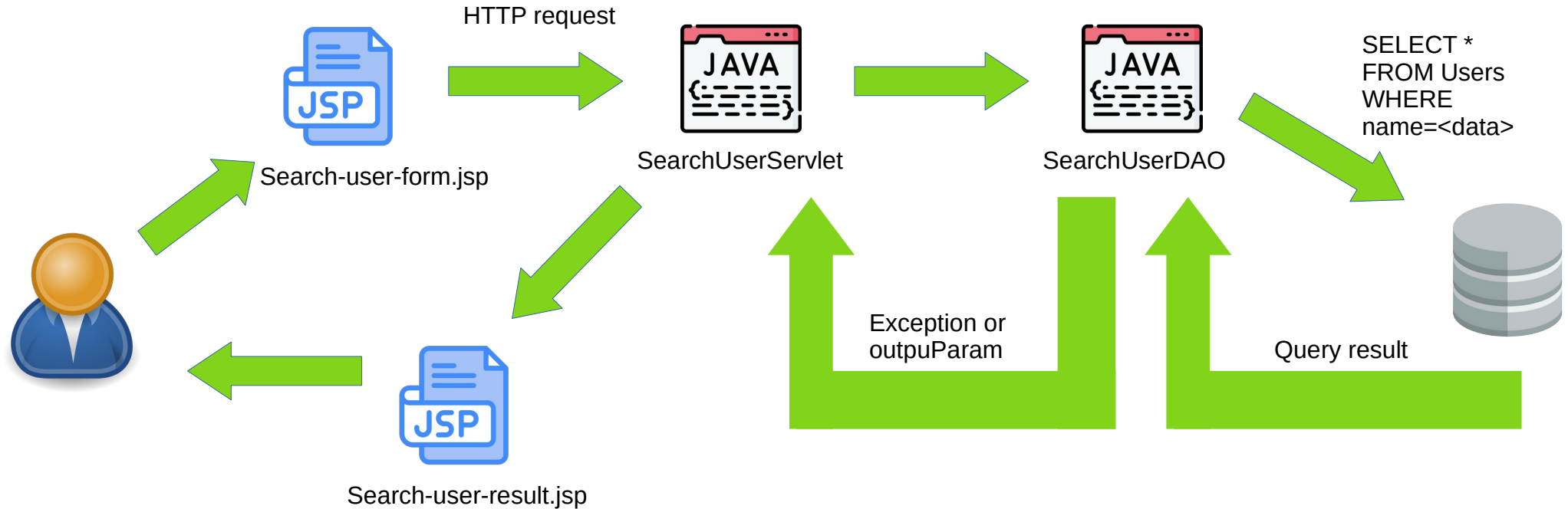
BookRec

User registration workflow:



BookRec

Searching for users workflow:



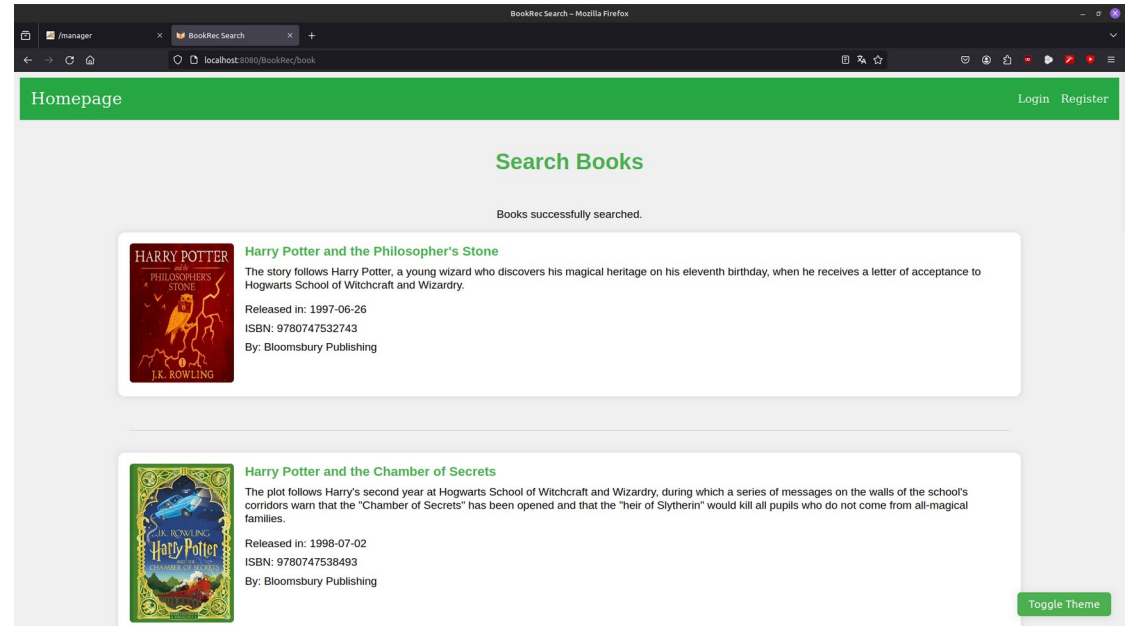
* for the search of books, another servlet is used to display image covers on the page

What about the design of the web pages?

We used mostly CSS, in order to obtain simple yet clear page for both results and form.

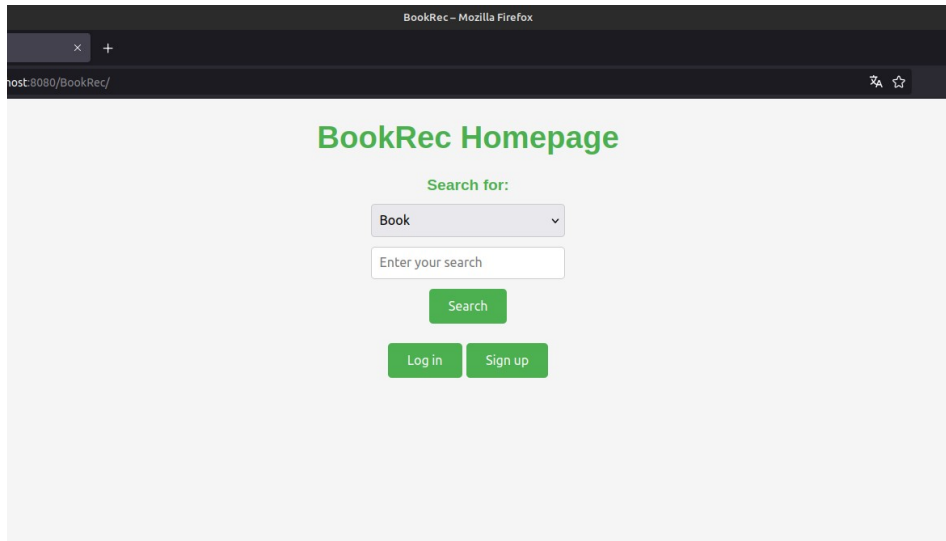


Following, some example of how CSS affects come of the pages in our website...

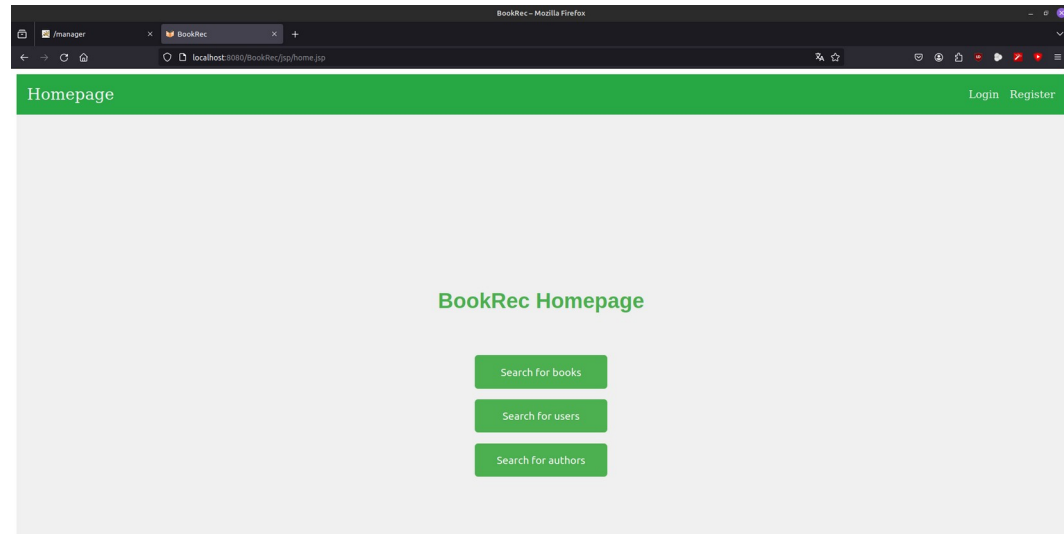


BookRec

Differences in homepage



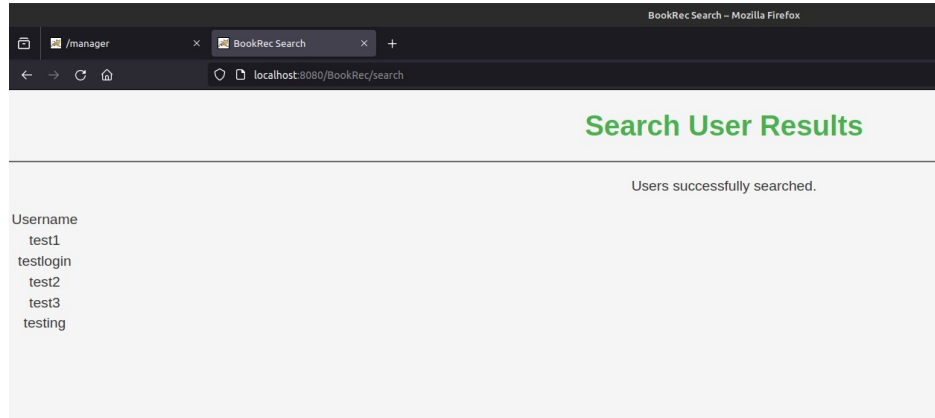
Without CSS



With CSS

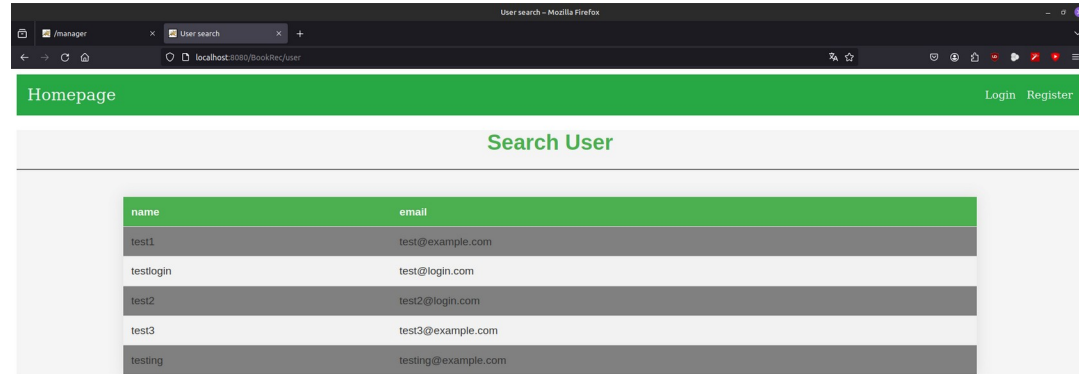
BookRec

Differences in user search results



Without CSS:

- No centering
- No alternate color
- No tabulation

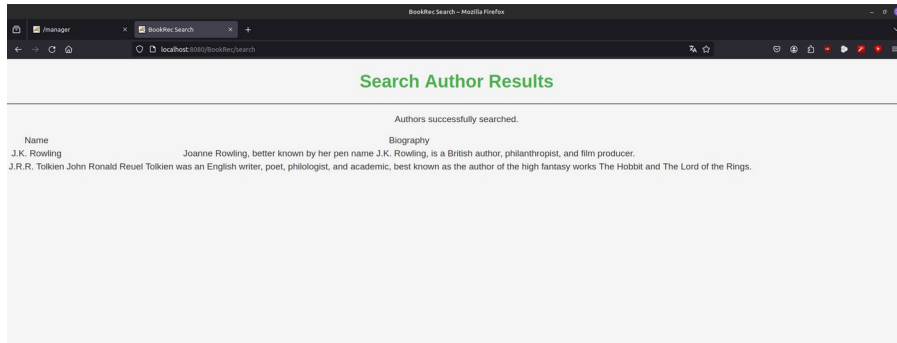


With CSS:

- Results are centered
- Correctly displayed in table form
- Alternate color for each line

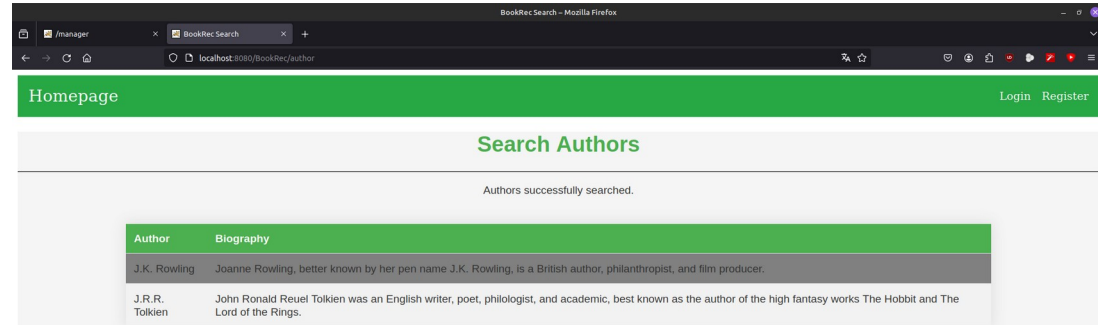
BookRec

Differences in authors search results



Without CSS:

- No centering
- No alternate color
- Basic tabulation



With CSS:

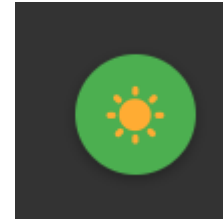
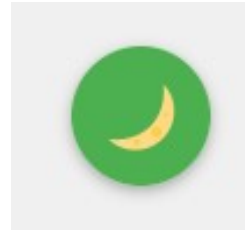
- Centered results are
- More readable table
- Alternate color for each line

BookRec

Some JS features:



The website has a light theme and a dark theme available on all pages, easily settable with a floating button



The navbar has a text-like link which is converted to POST request for the logout of users

Logout (test2) Delete account

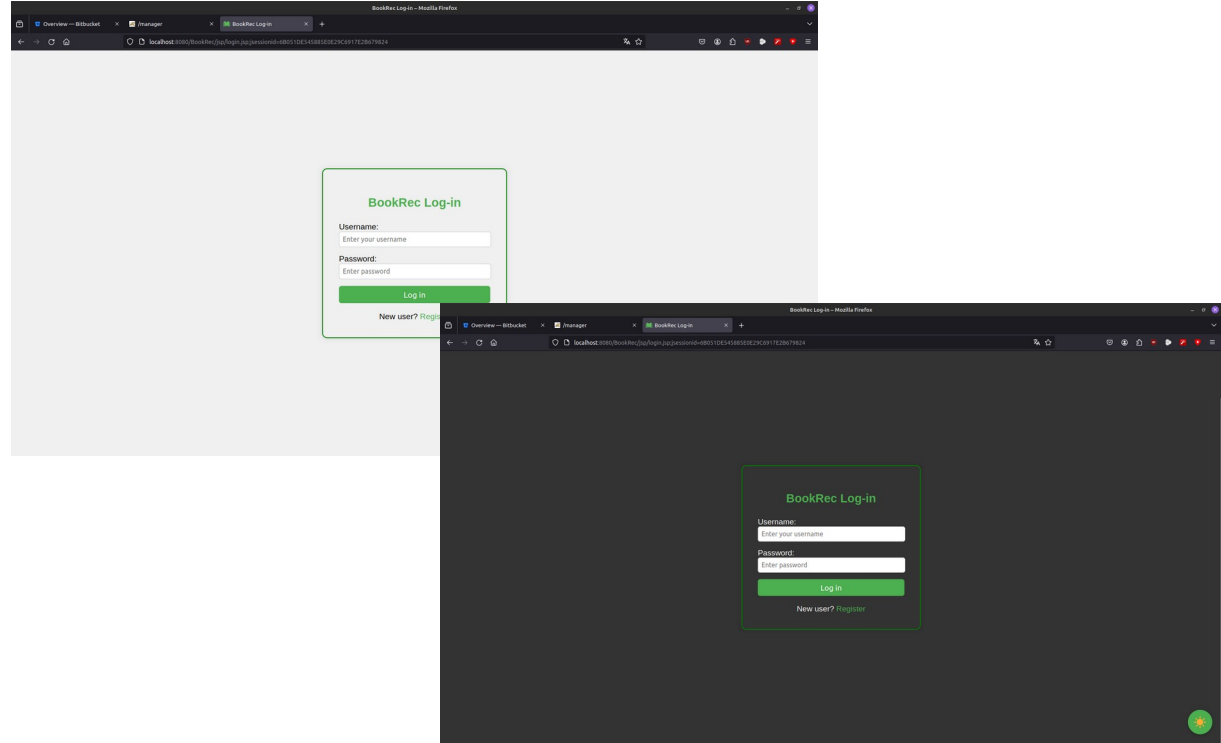
```
<li class="navig-item">
  <a class="navig-link" href="#" onclick="LogoutUser('<c:url value="/logout"/>'); return false;">Logout (
</li>
<li class="navig-item">
  <a class="navig-link" href="<c:url value="/jsp/delete-user.jsp"/>">Delete account</a>
</li>
```

BookRec

Dark theme:

```
theme8.js
1 // Function for dark theme
2 function activateDarkTheme() {
3   document.documentElement.setAttribute('data-theme', 'dark');
4   document.getElementById('theme').textContent = '🌑';
5   localStorage.setItem('theme', 'dark'); // Store the theme
6 }
7
8 // Function for light theme
9 function activateLightTheme() {
10  document.documentElement.removeAttribute('data-theme');
11  document.getElementById('theme').textContent = '🌞';
12  localStorage.setItem('theme', 'light'); // Store the theme
13 }
14
15 // Check if a theme was already set
16 function checkThemePreference() {
17   const preferredTheme = localStorage.getItem('theme');
18   if (preferredTheme === 'dark') {
19     activateDarkTheme();
20   } else {
21     activateLightTheme();
22   }
23 }
24
25 // Execute checkThemePreference at start
26 checkThemePreference();
27
28 // Add listener to the button
29 const themeToggleButton = document.getElementById('theme');
30
31 themeToggleButton.addEventListener('click', () => {
32   const currentTheme = document.documentElement.getAttribute('data-theme');
33   if (currentTheme === 'dark') {
34     activateLightTheme();
35   } else {
36     activateDarkTheme();
37   }
38 });
```

theme8.js

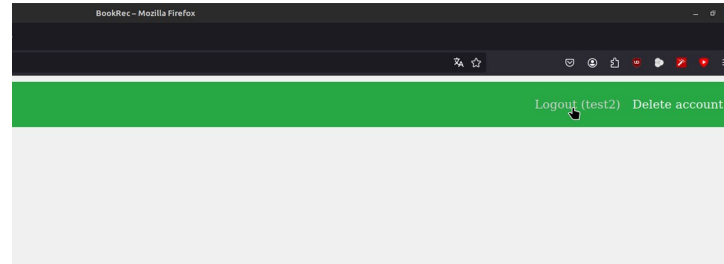


BookRec

Navbar link to form:

```
50 <script>
51 function LogoutUser(subURL) {
52     // Create form to send to LogoutServlet
53     var form = document.createElement("form");
54     form.method = "POST";
55     form.action = subURL;
56     // Adds the form to the document
57     document.body.appendChild(form);
58     // Submit form
59     form.submit();
60 }
61 </script>
```

Inline script, required to be properly loaded when link changed in the webapp

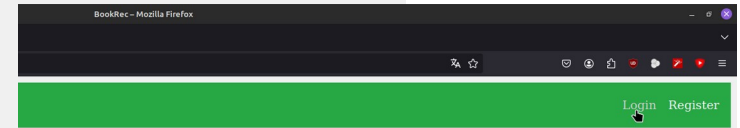


BookRec Homepage

Search for books

Search for users

Search for authors



BookRec Homepage

Search for books

Search for users

Search for authors

The Future of BookRec:

Though BookRec remains a college project for now, the journey has been incredibly rewarding. We've built a web application which can be expanded with a lot of features and refined to very elegant, responsive and interactive design.

Any Questions



