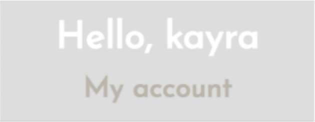
User accounts, reading lists, and personalised recommendation system

Personalisation is a theme in web development which has grown significantly in importance to user experience and functionality over time, so much so that it is a necessary element of nearly every website on the internet.

Personalisation helps a business to gain relevant insights into how a user navigates through their website, and can be a crucial element in improving the success of your business. As such, we felt it important that our website provides special features and personalisation techniques specific to individual users of our web application. In order to provide a user in session with relevant personalisation we began by developing a personal account for each user. A user's account then goes on to include special features such as a user's personal reading list as well as recommended books based on their preferences.

Beginning with design features, when a user enters our web application and register, they are greeted with a message in the navigation bar which is only displayed for logged in users. The message displays "Hello, [user_name]", as follows:

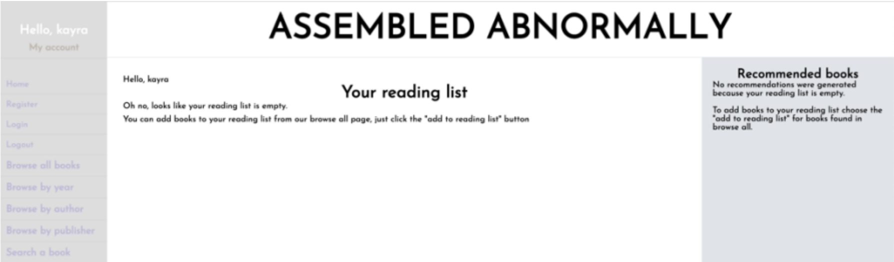


Hello, kayra

In this example the user is signed in under the user name "kayra".

Furthermore, being signed in enables the user an additional option in the navigation bar – below the greeting message – which provides them with access to their own personalised account.

When a user chooses to enter their account they are again greeted with their username and are able to view a personal reading list (centre of the web page), as well as a personal recommendation system of books (right hand side)



ASSEMBLED ABNORMALLY

Hello, kayra

Your reading list

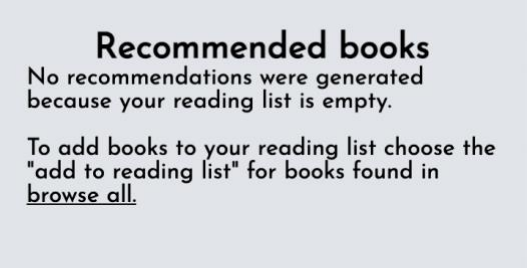
Oh no, looks like your reading list is empty.

You can add books to your reading list from our browse all page, just click the "add to reading list" button

Recommended books

No recommendations were generated because your reading list is empty.

To add books to your reading list choose the "add to reading list" for books found in browse all.

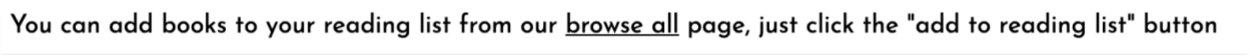


Recommended books

No recommendations were generated because your reading list is empty.

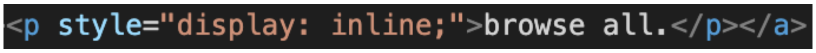
To add books to your reading list choose the "add to reading list" for books found in browse all.

Your account will display any books that you have added to your reading list, or, if in the case that you are either a new user (in a new session) or have not added any books to your reading list yet, then a message is displayed saying so, as well as explaining how to use the reading list functionality – with links included to direct a user to the correct place.



You can add books to your reading list from our browse all page, just click the "add to reading list" button

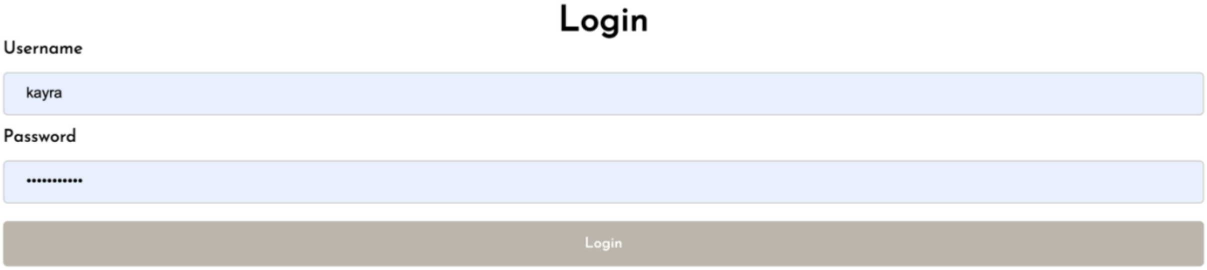
In the use of hover effects on buttons, we are able to distinguish between common text/information and objects which can be interacted with. In this such example, we felt that having the “browse all” button in line with the information about how to add books to your reading list was important to help the user understand the meaning of the object, as such we added some styling to the HTML (rather than the CSS) which keeps the line of information on the same row, then implementing the hover styling in the CSS.



```
<p style="display: inline;">browse all.</p></a>
```

Similarly, in the recommended books function, there is further information about how to allow recommended books to perform for you personally, and thus we included links to respective pages as well as corresponding information.

If a user chooses to follow the link to “browse all”, or rather accesses the “browse all” page from the main navigation, they will be able to view an “Add to reading list” button which is a feature only logged in users will be able to use successfully. To implement an authentication require functionality, we used a similar process as that to our reviews forms wherein we call the @login_required decorator to the respective blueprints. In this, if an un-logged-in user clicks the button to add a specific book to their reading list, they will be redirected to the login page. After which a user has successfully logged in, they will then be able to access the reading list features.



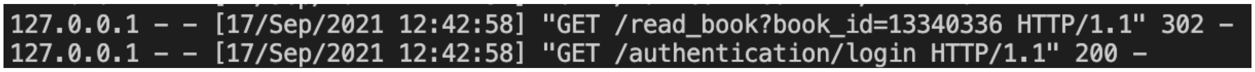
Login

Username

kayra

Password

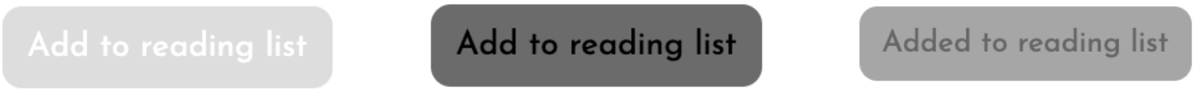
•••••••••••



```
127.0.0.1 - - [17/Sep/2021 12:42:58] "GET /read_book?book_id=13340336 HTTP/1.1" 302 -
127.0.0.1 - - [17/Sep/2021 12:42:58] "GET /authentication/login HTTP/1.1" 200 -
```

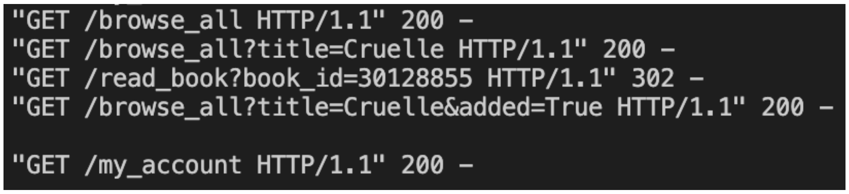
In the request/responses we can view the implementation of this, as we have the 302 (redirection) code requesting that a user logs in before attempting to use any such special features. The 200 (success) code displays that the authentication form (login form) has been successfully retrieved.

Moreover, once a user has added a book to their reading list, a new “added” tag is appended to the URL, telling the template that we can now disable the “Add to reading list” button. This implementation is made known to the user through design techniques which provide the “Add to reading list” button with the three different states:



the first two states display the button and the button hover. In these states a user can recognise they can interact with the button. In the last state, the button is disabled, and the text is changed to correspond to the state. In these such details, a user can navigate through the website more successfully and understand the functionality of the website.

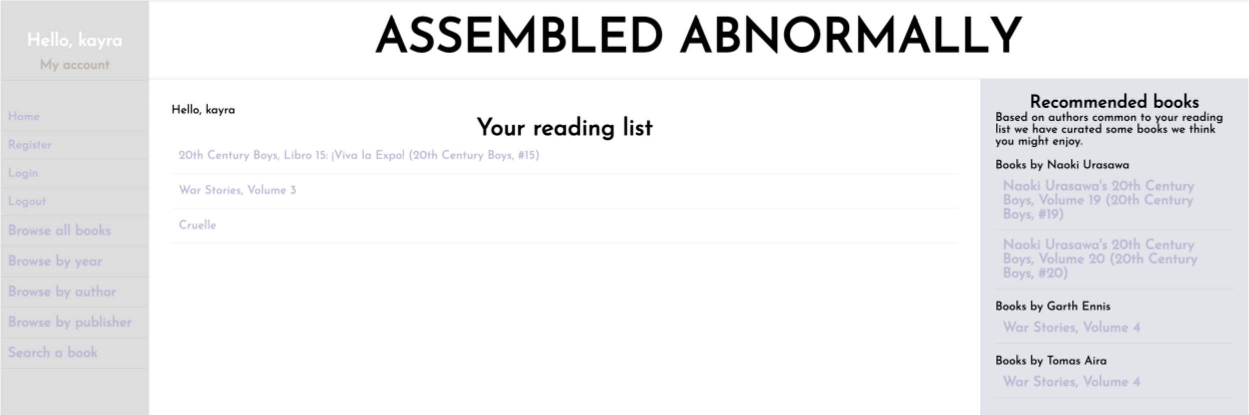
when a user does interact with the “Add to reading list” buttons for a particular book, the browser is directed to the blueprint which adds the book to the user’s personal reading list (internally) and then redirects the user back to the page they were currently on. This way, a user can continue browsing – rather than being redirected to their account to view their reading list. In this, we are promoting the user to continue looking for more books they may like – rather than halting this process. this technique can be helpful in the future if we wanted to develop our application to function more similarly to an e-commerce website for example. We can see this functionality in the responses below: 302 redirect to the current page, and 200 successfully adding books to user’s account.



```
"GET /browse_all HTTP/1.1" 200 -
"GET /browse_all?title=Cruelle HTTP/1.1" 200 -
"GET /read_book?book_id=30128855 HTTP/1.1" 302 -
"GET /browse_all?title=Cruelle&added=True HTTP/1.1" 200 -

"GET /my_account HTTP/1.1" 200 -
```

If a user adds some books to their reading list, all information for a user is stored in the current session – if you log in/out in the same session you can see the same stored reading list that you added in that current session, under that specific user. For example, below we can see that user “kayra” has added three books to her reading list. Each book in reading list is additionally accessible by a link to its corresponding page in “browse all”. This way, a user could quickly access the book’s individual/main page and perform features such as adding a review.



ASSEMBLED ABNORMALLY

Hello, kayra

Your reading list

20th Century Boys, Libro 15: ¡Viva la Expo! (20th Century Boys, #15)

War Stories, Volume 3

Cruelle

Recommended books

Based on authors common to your reading list we have curated some books we think you might enjoy.

Books by Naoki Urasawa

Naoki Urasawa's 20th Century Boys, Volume 19 (20th Century Boys, #19)

Naoki Urasawa's 20th Century Boys, Volume 20 (20th Century Boys, #20)

Books by Garth Ennis

War Stories, Volume 4

Books by Tomas Aira

War Stories, Volume 4

Figure 1

Moreover, the special book recommendation system provides recommended books for a user based on the books they have added to their reading list. The recommendation system works by checking the authors of books contained in a user’s reading list and returning other books by the same authors, that the user has not yet read. In this, if a user has read all books by a specific author, then that author will not show up in the recommended books. As seen in *fig. 1*, user “kayra” added the book Cruelle, by Florence Dupre la Tour, yet this author is not listed under her recommended books as she has already read all books by Florence Dupre la Tour. Similarly to the reading list links, we have provided links to all of the recommended books, thus allowing the user quick access to the book’s individual page wherein the user can click “Add to reading list” if they decide to.

Additionally, we made the decision to include our accessibility features such as the “Add to reading list” and “review” buttons only in the browse all page. this design choice was made in order to provide a simple navigation through the technical functionalities our website provides. For users to be able to interact with the books successfully it makes sense to provide a simple and clear accessibility to such special features. We found that it only made sense to have such features available in the “browse all” page as this is the only place which allows each book to be viewed individually, as the books here are stored based on their title (a unique feature to each book). For example the “Browse by author” or “browse by publisher” pages contain one or more book per page, and it would not make sense for a user to add reviews on an author or publisher, respectively – likewise with adding books to a reading list, we thought it would make the most sense to have the website function in this such way so they can only add books to their reading list; not authors and publishers for example.

In such use of special features, we were able to provide a tailored experience for a user in session on our web application. Such techniques allow further development toward a successful application which can promote specific features of a website which you would like to be interacted with – for example if you were planning on developing your application into an e-commerce based website, then, having implemented reading list functionality, you are able to simply modify it to be a cart of books the user would like to purchase for example.