Web Application with User Authentication and Bookmarking

Fahim Ferdous

ID: 1911631042

North-South University

CSE 299.2, Junior Design

Instructor: Ziaul Hossain

5/13/22

Abstract

A Web application with user authentication and personalization was implemented with custom functions. The implementation were done using PHP and Mysql and hosted on a local network. Bookmarking of URLs was used to implement User personalization.

Keywords: Web application, Authentication, Personalization, Bookmarking, URL

INTRODUCTION

This project was built on understanding the workings of user registration, authentication, user personalization through content tracking behavior, and using the result to generate content to their preferences.

BACKGROUND

This project allows users to create a set of online bookmarks and proposes other sites they might be interested in depending on their previous activity. User customization, in general, may be utilized in practically any web-based application to offer users the material they want in the manner they desire.

METHODOLOGY

This project began with examining a set of criteria comparable to those received from a client. Development of those requirements into a collection of solution components, creation of a design to connect those components, and implementation of each element.

The following functionality was implemented:

■ Logging in and authenticating users

■ Managing passwords

■ Recording user preferences

■ Personalizing content

■ Recommending content based on existing knowledge about a user

Solution Components :

This system should allow users to log in and save their own bookmarks, as well as receive recommendations for additional sites to visit based on their particular tastes.

These fall into three main categories:

■ Identify individual users, and have some way of authenticating them.

■ Store bookmarks for an individual user. Users should be able to add and delete bookmarks.

■ Recommend to users sites that might appeal to them, based on their known preference.

User Identification and Personalization

For user authentication, there are several options. To associate users with certain customization data, their logins and passwords were saved in a MySQL database and authenticated against it.

To let users log in with usernames and passwords, the following components were implemented:

■ Users should be able to register their usernames and passwords, some restrictions on the length and format, and in an encrypted format for security reasons.

■ Users should be able to log in with the details they supplied in the registration process.

■ Users should be able to log out after they have finished using a site. This capacity is not very necessary if individuals access the site from their own computer, but it is critical for security if they access the site from a shared computer, such as one in a library.

■ The site needs to be able to check whether a particular user is logged in, and then access data for the logged-in user.

■ Users should be able to change their passwords as an aid to security.

■ Users should be able to reset their passwords without needing personal assistance.

For purposes of this project, custom functions were written in PHP for all of these pieces of functionality. Most of them will be reusable, or reusable with minor modifications, in other projects.

Storing Bookmarks

To store a user's bookmarks, some relational tables were set up in MySQL database.

Functionalities:

■ Users should be able to retrieve and view their bookmarks.

■ Users should be able to add new bookmarks. The site should check that these are valid URLs.

■ Users should be able to delete bookmarks.

Again, custom functions were written for each of these pieces of functionality.

Recommending Bookmarks

A "like minds" recommendation system that looks for people who have the same bookmark as an existing logged-in user and promotes their additional bookmarks to the user. Only bookmarks maintained by more than one other user will be recommended to avoid proposing any personal bookmarks.

Solution Overview

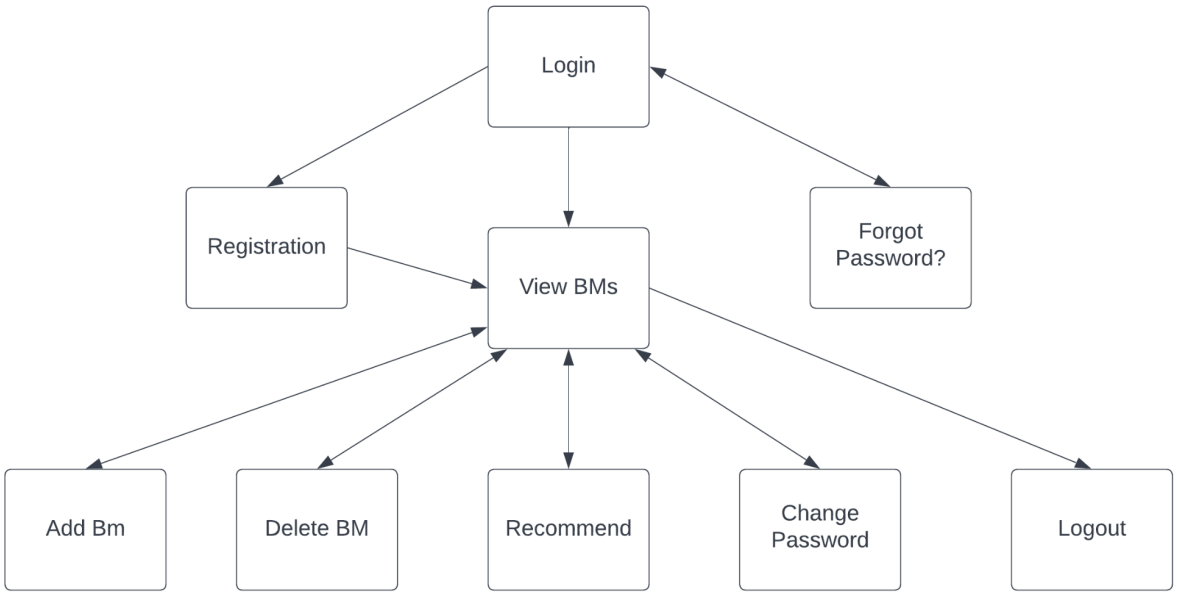

Fig: The possible paths through the system

Results

After user registration, and authentication they are transferred to the home page with a menu of options and any bookmarks they have added.

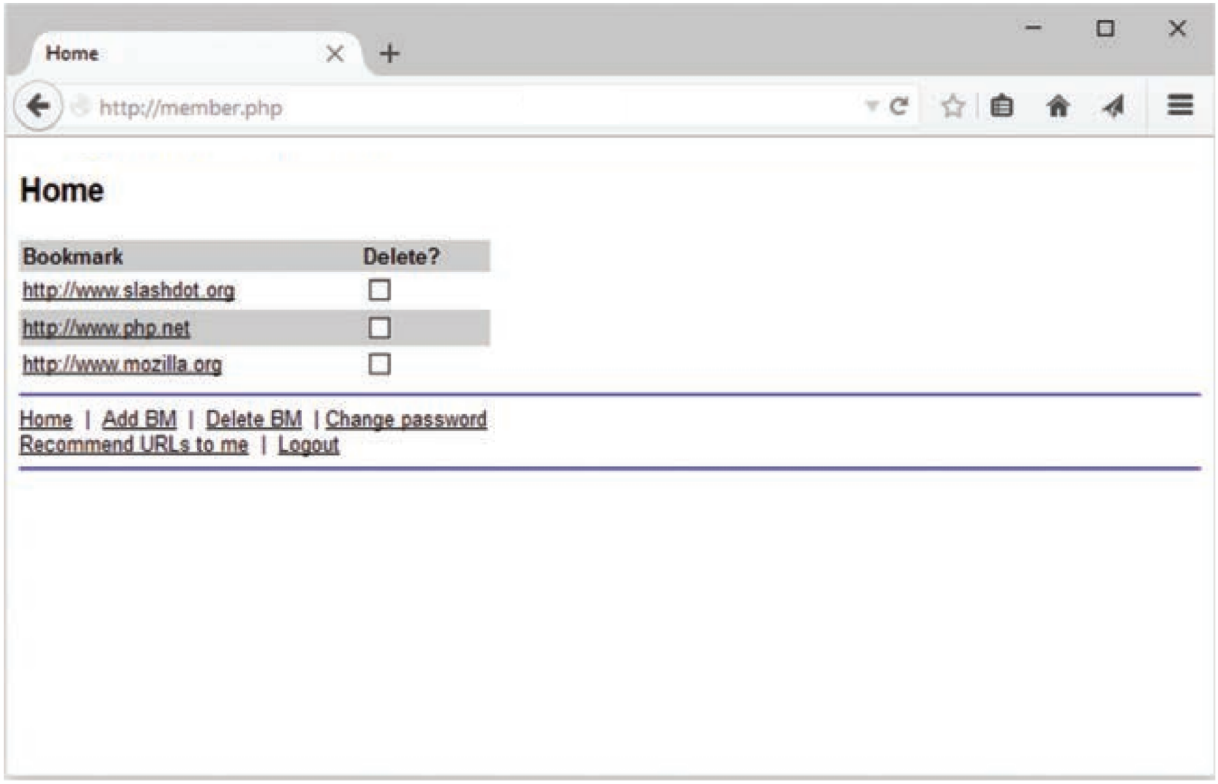

Fig: Output of member.php script

The add BM menu option presents them with a page that allows them to save bookmarks.

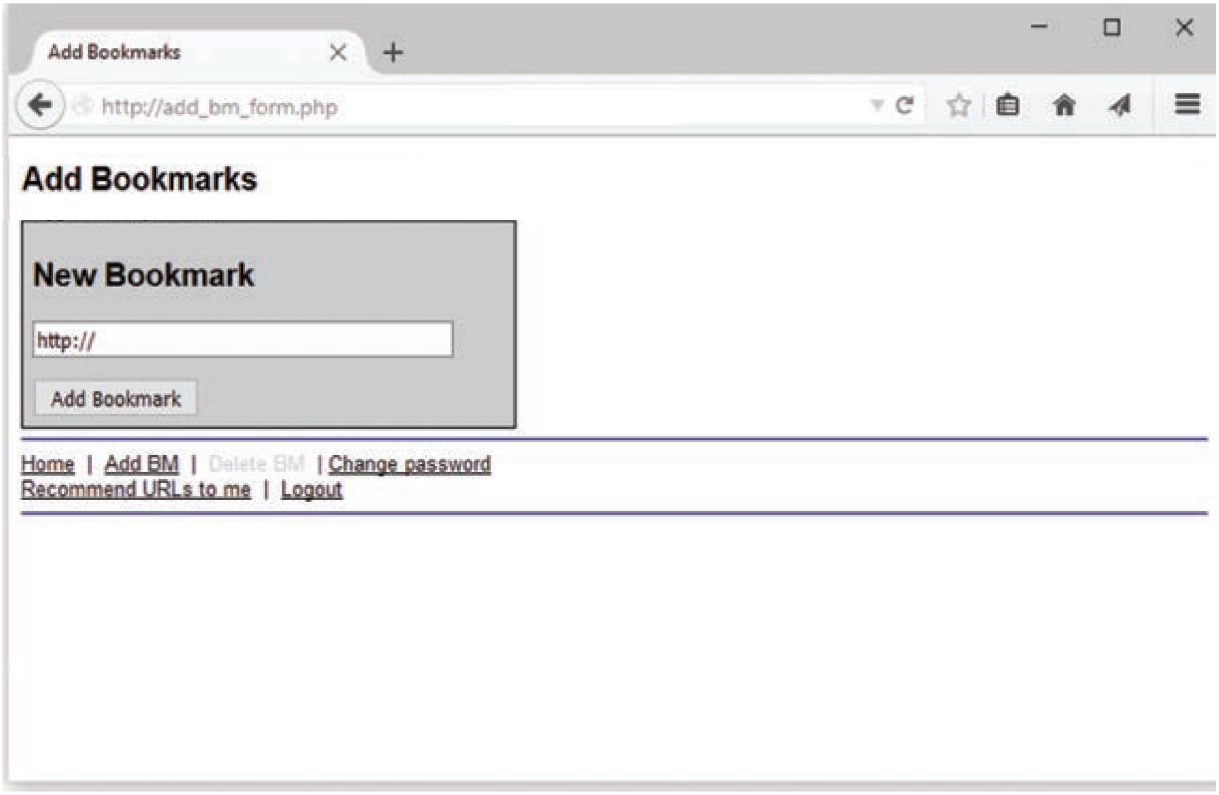

Fig: Output of The add_bm_form.php script supplies a form where users can add bookmarks to their bookmark pages.

When some bookmarks are marked for deletion and clicks on the Delete BM link on the menu, the form containing the URLs is submitted.

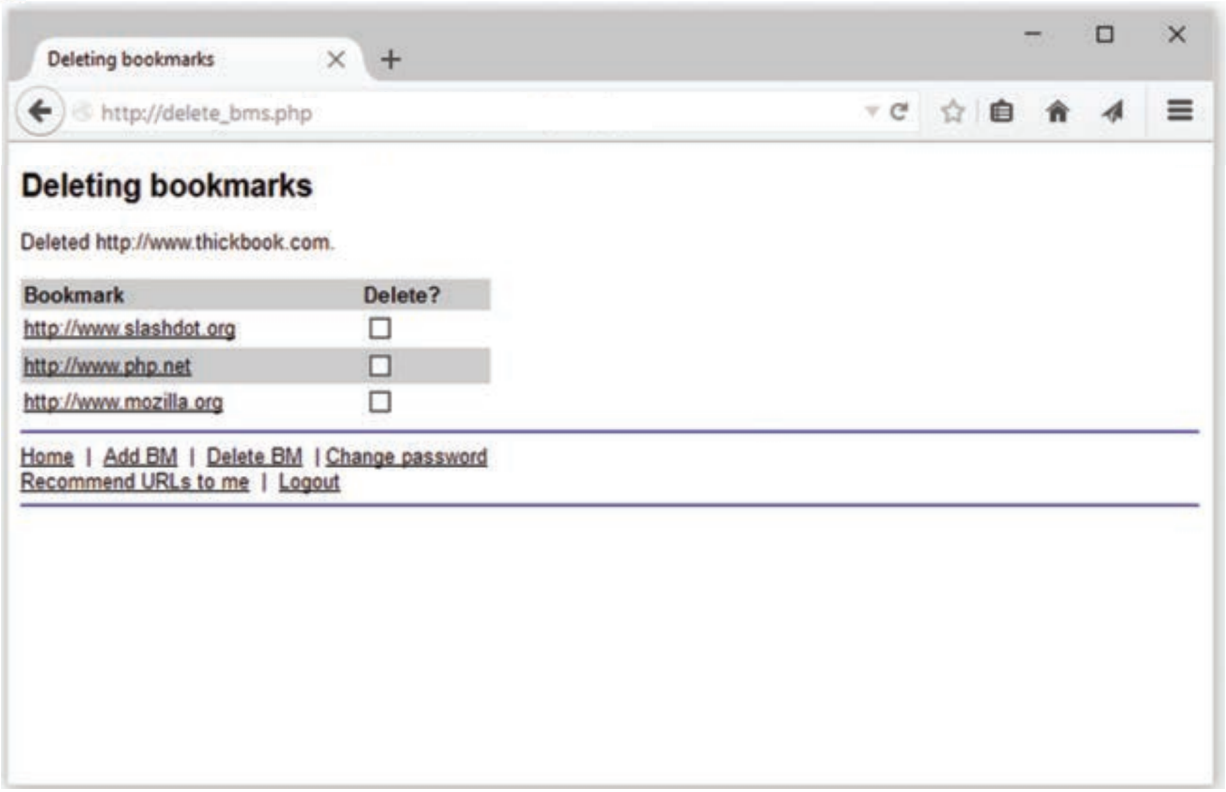

Fig: Output of The deletion script notifies the user of deleted bookmarks and then displays the remaining bookmarks

The “ like minds” recommendations approach was used to fetch other bookmarks used by other users in the database.

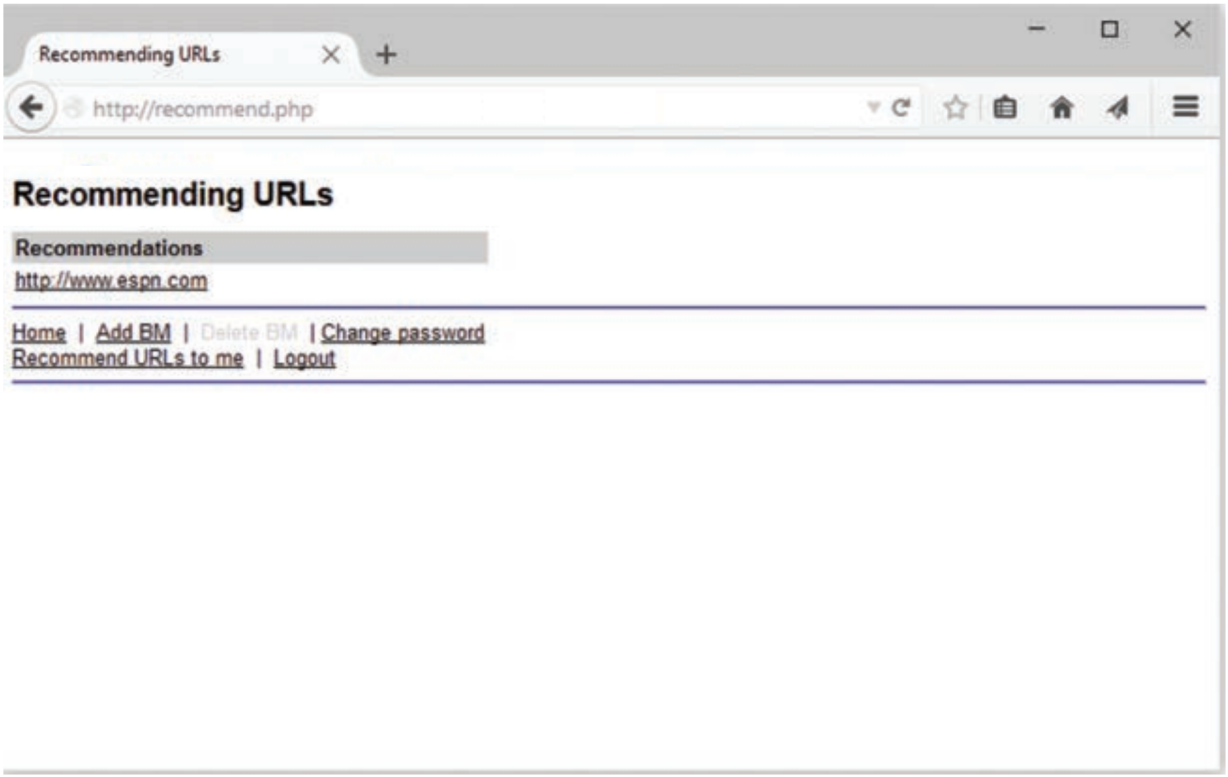

Fig: Output of The recommend.php script has recommended that this user might like espn.com. At least two other users in the database who both like espn.com have this site bookmarked

Discussion

The basic functionality of the system application has been implemented with custom functions written with reuse, scalability, and with any necessary modifications in mind without damaging the code structure.

There are many possible extensions. Some considerations may be :

■ A grouping of bookmarks by topic

■ An “Add this to my bookmarks” link for recommendations

■ Recommendations based on the most popular URLs in the database or on a particular topic

■ An administrative interface to set up and administer users and topics

■ Ways to make recommended bookmarks more intelligent or faster

■ Additional error checking of user input

Conclusion

There might be more efficient ways of implementing the system by the use of tech stacks since writing custom functions require intensive time and effort and thinking whereas tech stacks come with built-in and more efficient implementations of the functionalities. Nevertheless, the insight gained from building from scratch cannot be achieved by using ‘off-the-shelf’ integrations.

Recommendations

With the advent of WEB 3.0, the insight may prove useful in designing the system for WEB 3.0 since only a few frameworks are available. Such an implementation is highly recommended for the next point of departure.