Project: A digital library search engine for electronic thesis and dissertations

Student name: Triveni Sangam Saraswathi Edla (UIN:01160800)

Github link: https://github.com/Triveniedla/webprogramming-Triveniedla

1. Overview

Nowadays people are using electronic thesis and dissertations instead of hard copies. With current pandemic and social distancing, there is increased usage of online resources for electronic copies. In this project, a website is designed which provides a digital library search engine for electronic theses and dissertation. Laravel 8.x framework is used to build the website. Laravel is a free, open source PHP web framework for web applications. The Laravel framework is conned to MySQL databases to store and retrieve the data. A combination of php, JavaScript are used to design the website. The basic specifications are met in the designed website and along with one optional requirement of advance search option.

Table 1: Overview of status for Milestone 1 specifications.

Fulfilled	#	Description
Yes	1	The website should provide a search box at the landing page
Yes	2	There should be a search button next to the search box;
Yes	3	Users must be able to register new accounts using email addresses;
Yes	4	Password must be encrypted before storing in the database;
Yes	5	Users cannot register duplicate accounts using the same email address
Yes	6	Users should be able to log into your website using the accounts they registered
Yes	7	Users should be able to reset their passwords if they forget it;
Yes	8	The user login process must use the HTTP POST method;
Yes	9	User information shall be stored in a MySQL database;
Yes	10	The website should have a homepage for each user, where they can view their profiles, change passwords, and update information.

Figure 1: This is my search engine front page.

2. Database Design

The database is created in MySQL with the name and the tables in the database is created using migrations using "php artisan migrate" command. The migrations create several tables, and the "users" tables is used to store the details of the users. The contents of the database is shown in Table.1. The id is a unique value automatically created by the database when new users are entered, and it is unsigned value. The email is unique value and users cannot create multiple accounts with same email id. The Email_verified_at token contains the timestamp when the email was verified. The password is encrypted and remember_token is encrypted. Remember token is used to store a token for remembering the user's session. The timestamp at which the user was created is stored in created at and last access of the account is stored in updated__at field. The database also contains the password_resets tables which stores the details of the user email who have requested for password reset, the reset_token and timestamp at which the request was created.

Table 2: Database design.

Field	**Type**	**Key**	**Example**
Id	Bigint unsigned	Primary	1
name	VARCHAR(30)		triveni
emaill	VARCHAR(30)	UNI	tedla001@odu.edu
Email_verified_at	timestamp		1990
password	varchar(255)		Encrypted
remember_token	varchar(100)		Encrypted
Created_at	timestamp		2020-10-05 07:55:08
Updated_at	timestamp		2020-10-06 02:58:52

3. Website architecture

Figure. 2 shows the architecture of the webpage that is designed. The landing page contains the links of login page, user registration page. The user can register an account using emailed, password and password need to be repeated. When the user is registered, it is redirected to user home page and the user session starts. The Login page gives an option to login into their account using email, password and the page reroutes to the user home page once login with correct credentials. Login page also provides option to reset password by sending an email to the users' account if the user wants to reset the password. The user home page contains the search bar with advance search options and logout and personal details page. In the personal's details page, the user can change their passwords by sending an email the users' email.

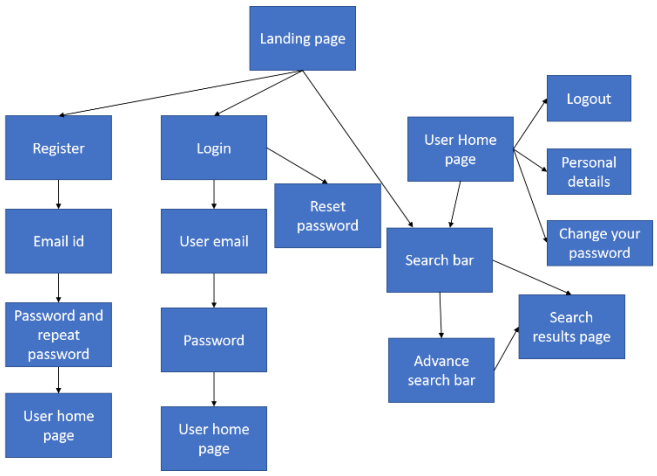

Figure 2: website architecture design.

4. Frontend design

All the pages shown in Fig. 2 is created using mix of html, java script and css files and the home page also contains a search function. The pages for the website are created in view folder and they are routed using routes and controller. The controllers use middleware to authenticates new users and users that are logged in. Once the authenticated user is logged in, the route middleware gives the users access to a given route. The passwords are encrypted using Hash option available in Laravel and stored in the database. Models is used to access the user tables in the database and used to store the new user registration into database. Model checks if the duplicate user is registered by checking the data in database users table and gives an error if the user is already registered. The data of the user is saved into the database once registered. The Post method is used to login into the user account. The profile page and logout page is added to the user page as dropdown. Using the profile page, the users can update and reset the password. Larval provides many tools to automatically configure and "composer require Laravel/ui" is used to create basic views and scaffold all the views and routes using controllers. The initial views and controllers are modified to the ETD application.

5. Other features to be implemented:

The feature to get an confirmation email to verify the email addresses is not implemented will be implemented in the next mile stone.