

WDD PACE University of Winnipeg

2022-May-3

Lihang Yao

Proposal for

Gardener Website Upgrade



Table of Content

TABLE OF CONTENT	2
1. INTRODUCTION.....	3
2. FEATURES AND FUNCTIONS	4
1.1 For normal users.....	4
1.2 For Admin	4
3. DATABASE	5
1.1 users table	5
1.2 posts table	5
1.3 comments table	5
1.4 categories table	5
1.5 log table	5
4. USE CASES.....	6
1.7 For unauthenticated users	6
1.8 For authenticate users.....	6
1.9 For Admin	6
5. SECURITY	7
6. SITEMAP	8
1.10 Front-end Sitemap.....	8
1.11 6.2 Back-end Sitemap	8
7. LOGIN AND REGISTER.....	9
8. PHYSICAL DATA MODEL	10
9. TIMETABLE	11
10. SITE BIBLE.....	12

1. Introduction

Gardener website is a local platform for people to interact with each other. Now the owner of website decides to upgrade the website to a dynamic changing personal blog website instead of a static website for exhibition purpose.

To meet the requirements to upgrade the current website, our company will take the responsibility to apply a new technique on this project.

We will use PHP as the primary programming language, HTML to construct the fundamental infrastructure, CSS to present a gorgeous style and JavaScript to enhance the functionality of the current website. In addition, we will use MySQL as the database for this project.

We will use MVC to dispatch every user request, by this way, we can make the program more organized and easy to maintain.

2. Features and Functions

2.1 For normal users

- normal user can login and logout
- normal user can register an account
- normal user without login can review all posts
- normal user with login can leave a comment on posts
- normal user can view all comments they left on the website
- normal user can delete their comments
- normal user can view other user's comments
- normal user can search for specific posts
- normal user can view posts by categories

2.2 For Admin

- Admin can do anything as a login normal user
- Only Admin can login to the dashboard of the administration system
- Admin can add a new post
- Admin can edit existing posts
- Admin can delete any posts
- Admin can view lists of current posts, users, comments, and categories

3. Database

There are five tables in the database. Every table will have a unique id field as primary key to target an individual row. Each table have time related fields to record the time of creation and update. Tables with a 'deleted' field will determine whether this record is deleted.

- users table

Users table will collect user information on first name, last name, mailing information, phone, email, and users must create a password for login purpose.

- posts table

Posts table contains the title, author-id, summary, content, image, category, status of post and privilege to allow comment.

- comments table

Comments table contains the information of the comments. There will be post-id field relating to the post where this record of comment resides.

- categories table

Categories table contains names of five categories of posts.

- log table

Log table stores the records of all users' entries.

4. Use Cases

- For unauthenticated users

Unauthenticated users can view all posts and comments that other users left on the website. But if they want to leave a comment on a post, they must register an account first. On register page, they must input all required fields and click submit, then it will bring the user to profile page if registered successfully.

- For authenticate users

Authenticate users could browse all posts on the website after login. And they can leave a comment on any allowed-comment posts. If they enter their individual profile, they will find all comments they left previously.

- For Admin

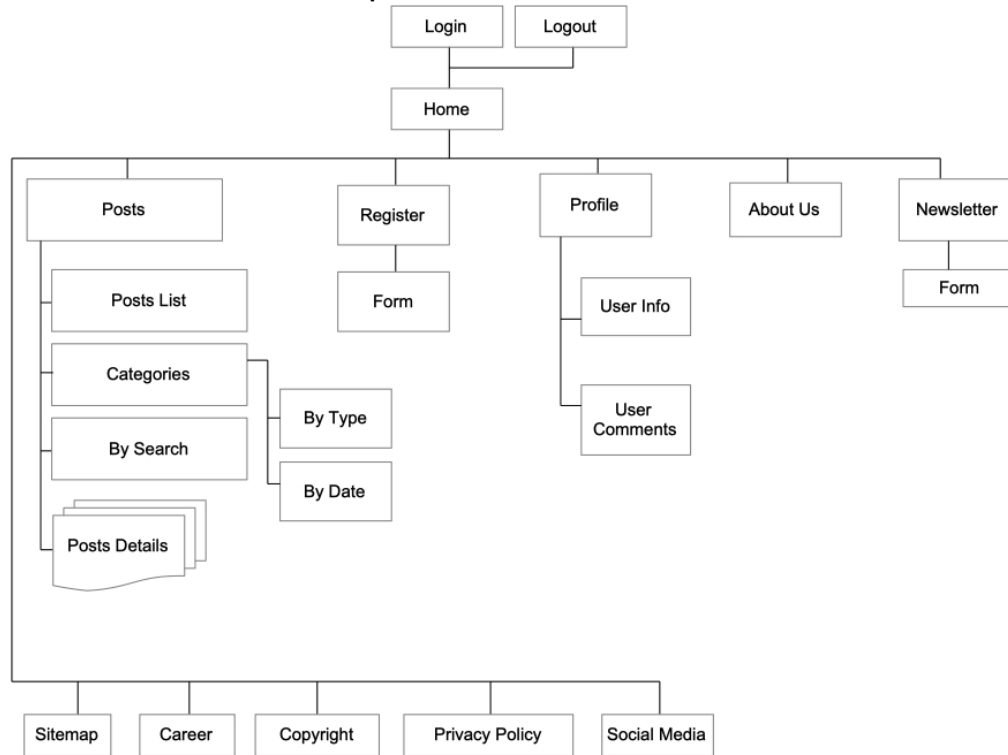
Admin can do anything what an authenticate user can do on the website. Besides, Admin can login the dashboard to access the administration system. They can view list all posts, comments, and user information. And they can create, update, and delete any posts. On user page, Admin can delete or update any user's information. On comments page, Admin can delete any comments if possible.

5. Security

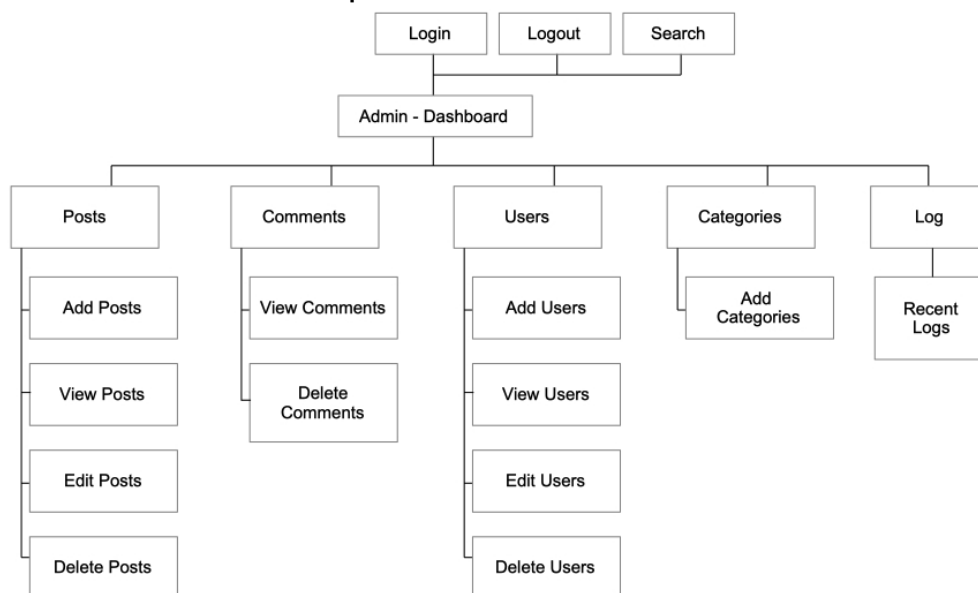
- To prevent XSS attacks, we will use PHP built-in function `htmlspecialchars()` to escape any output to users' web browser. Besides, we will validate any value of form field from user's input.
- To prevent SQL injection attacks, we will use parameters to bind the value with the placeholders. By this way, we can prevent any malicious code will be injected into our databases.
- To prevent CSRF attacks, we will use csrf token to insert into a hidden field in every form. We will validate the csrf token every time on a POST request method. If the csrf token doesn't match with the one on our server, we will end this session immediately.

6. Sitemap

- Front-end Sitemap



- Back-end Sitemap



7. Login and Register

The screenshot shows a web browser window with the title 'Login' and the URL 'localhost/?p=login'. The page features the 'Gardener' logo and navigation links: Home, Post, About Us, and Newsletter. On the right, there are links for 'Login' and 'Register'. The main content area is titled 'Login' and contains a form with two input fields: 'Email' (containing 'admin@gardener.com') and 'Password' (masked with dots). Below the fields is a blue 'Submit' button. At the bottom of the page, there is a dark footer with a grid of links organized into four columns: Account, Timeline, Community, and Store. The 'Account' column includes links for User Name, Profile, Email, Password, Address, and Telephone. The 'Timeline' column includes links for My Items, My Lists, Events, Alerts, Reminder, and New Routine. The 'Community' column includes links for Top Trends, My Posts, Likes, Comments, Messages, and New Posts. The 'Store' column includes links for Orders, Cart, Returns, Shipping, Customer Service, and FAQ. Social media icons for Facebook, Instagram, YouTube, and Twitter are located in the bottom right corner of the footer.

Login

Email

Password

[Submit](#)

Account	Timeline	Community	Store
User Name	My Items	Top Trends	Orders
Profile	My Lists	My Posts	Cart
Email	Events	Likes	Returns
Password	Alerts	Comments	Shipping
Address	Reminder	Messages	Customer Service
Telephone	New Routine	New Posts	FAQ

The screenshot shows a web browser window with the title 'Register' and the URL 'localhost/?p=register'. The page features the 'Gardener' logo and navigation links: Home, Post, About Us, and Newsletter. On the right, there are links for 'Login' and 'Register'. The main content area is titled 'Register' and contains a form with the following fields: First Name, Last Name, Street, City, Postal Code, Province, Country, Phone, Email (containing 'admin@gardener.com'), Password (masked with dots), Input password again, and a checkbox for 'Subscribe to newsletter'. Below the fields is a blue 'Submit' button.

Register

First Name

Last Name

Street

City

Postal Code

Province

Country

Phone

Email

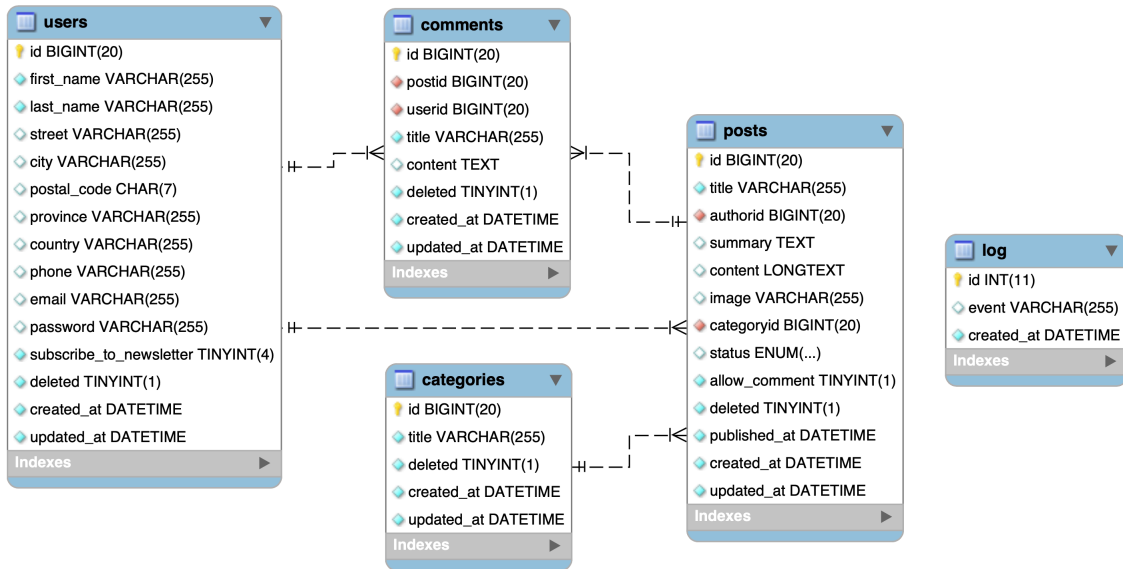
Password

Input password again

Subscribe to newsletter ☐

[Submit](#)

8. Physical data model



9. Timetable

Stage	Date
Kickoff meeting	May 2
Initial Wireframe	May 3
Design Mockup	May 7
Copywrite	May 7
Programming	May 31
Testing	June 2
Website Launch	June 15

10. Site bible

- **Hosting Server**

AWS EC2

- **PHP Version**

PHP 8.1.5

- **Database**

MySQL 10.4.21

- **IP of Service**

52.60.51.49

- **URL of Project**

<http://capstone.leon-web-dev.com>

- **Admin User Login**

Email: admin@gardener.com

Password: P@ssw0rd?

- **Sample Normal User Login**

Email: chris@gardener.com

Password: P@ssw0rd?