CVWO Blog Final Write Up

User Manual

About:
This is a weblog that provides basic blog facilities to writers and reader.

Types of users:
1. Readers
2. Writers
3. Admin

Services provided by the blog:
1. Read all the blog posts on the website.
   Link: '/'
   Who can access: Readers, Writers, Admin
2. Search for articles with specific tags.
   Link: **'/views/search.php'**
   Who can access: Readers, Writers, Admin
3. Create new posts and edit/delete old posts.
   Link: **'/views/portal.php'**
   Who can access: Writers, Admin
4. Comment on articles.
   Link: **(go to individual blog post)**
   Who can access: Writers, Admin
5. Create new user accounts for writers.
   Link: **'/views/signup_page.php'**
   Who can access: Admin
6. Reset the database.
   Link: **'/user/db_reset.php'**
   Who can access: Admin

Accomplishments

1. Basic understanding of the web.

Through much failure and research, I believe I was able to learn much about how the web functions. Especially what HTTP the request/response model it employs. I was also able to understand the role of servers and clients in a web application. In short, building a web app exposed me to many technologies that helped me better understand how the web works.

2. App development cycle

Through much iteration over trying to improve and refine the application, I experienced several sections of an application's development cycle. At first, I made many mistakes in creating my first blog in that, I focused on the specifics straight away. This way, I was trying to patch up my mistakes on by one. Over much iteration, I realized the need to become modular and design the application over at a bigger picture before delving into the specifics. I learnt to separate front-end from back-end work. I learnt to handle exceptions and consider many different use cases. I tried to make my code more organized and better documented.

3. Good database design

As I explored more ideas to implement such as comments and tags, I tried to look back at the bigger picture to see how I can store information more efficiently. Initially, I was thinking of creating a table for each user and creating a table for each article. But I soon realized that this would be a highly inefficient way to organize data in a RDBMS. I can use primary and foreign keys to easily map data elements across tables which will ensure data integrity as well as faster indexing. So I created four tables each storing different types of content. (Users, articles, comments, tags)

4. Good code organization

Although I haven't mastered this yet, I believe this assignment has made me more careful of handling code loosely. I realize it become easier to debug code when it is organized.

Another principle I haven't mastered yet but appreciated yet is DRY (don't repeat yourself). Although I haven't fully abstracted my code, I realized the redundancy in repeating code. Bad abstraction makes debugging harder (because it becomes harder to find the bug).

5. Data and Web Security

Probably one of the biggest things I have thought about and tried to implement in my web app. Every time I create a new feature or use case, I constantly think about whether it is secure or not. Whether it can be exploited by other users. For example preventing non writers from posting/editing/deleting. Also making sure that, writers cannot delete/edit other's articles made me think greatly about how to make my application more secure from exploits. Every time I made database requests, I tried to ensure that I use prepared statements instead of queries as queries are susceptible to attacks. I also limited the use of JavaScript code in my application as I believe it can be easily exploited by the user. This is why I relied on a lot more server side programming and validation to ensure greater security of the web app.

6. Front end design

Although I did not have much time to do extensive front end design, I tried to make the look very clean and intuitive. This was my primary focus. I paid attention to what would be excessive and what would be more intuitive in front-end design. I believe I was able to create a neat looking app with a clean and intuitive interface.