

Author

Name : Ujit Kumar

Roll Number : 21f3000786

Email : 21f3000786@student.onlinedegree.iitm.ac.in

About Me:

I am currently enrolled in the Diploma program in both Programming and Data Science, having successfully completed the foundational level of the course. My passion and interest in Data Science and programming has been evident throughout my studies and I am continuously seeking to enhance my knowledge and skills in these fields. Additionally, I am excited to apply my acquired knowledge to real-world challenges and contribute to the growth of the field.

Description

I have developed a Flask-based multiuser application, named 'Blogpost'. The application grants users the ability to easily create and publish their own blogs, complete with the option to incorporate images. Furthermore, users can seamlessly follow other bloggers and stay up to date with their latest posts.

Technologies Used

In the project I have used the below technologies:

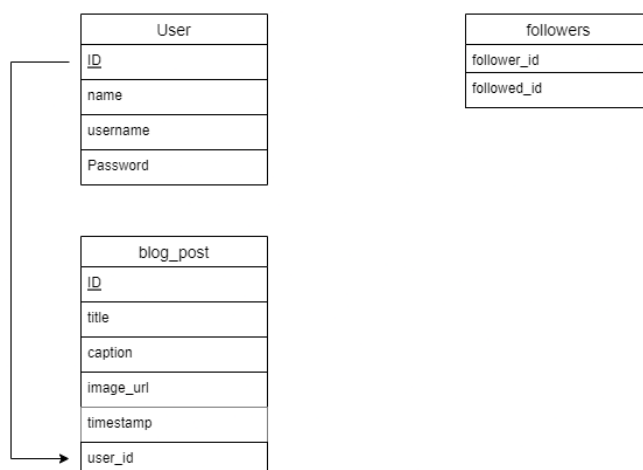
1. **flask** - Utilized flask to create a dynamic and robust web application in my project.
2. **flask_restful** - Implemented flask_restful to enhance functionality and create RESTful APIs
3. **flask_sqlalchemy** - Used Flask_sqlalchemy to effectively interact with databases(ORM)
4. **HTML** - Employed HTML to structure and create well-formed web pages.
5. **CSS** - Incorporated CSS to provide a polished and visually appealing user interface in my project.

Note:

Python is the main programming language used in the flask and flask_restful and flask_sqlalchemy and flask_sqlalchemy and is the foundation for the project.

DB Schema Design

The below diagram shows the Data Base Structure.



API DESIGN

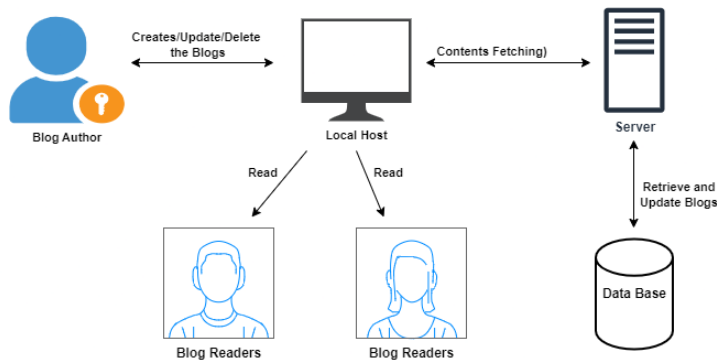
The CRUD (Create, Read, Update, and Delete) operations for both user and blogpost entities have been implemented and documented in the OpenAPI specification, utilizing the YAML format. The file containing the API definition is named "**blogpost.yaml**" for easy reference and management.

Server URL : http://127.0.0.1:5000/

Request Type	Endpoints	Description
GET	/signup	Used Returns the signup page
POST	/signup	Creates a new user
GET	/logout	Logout the user from the session
GET	/updateuser/{user_id}	Returns the update user page
POST	/updateuser/{user_id}	Updates the user information
GET	/deleteuser/{username}	Deletes the user
GET	/profile/{username}	Returns the user's profile page
POST	/createpost	Creates a new blog post
GET	/createpost	Returns the create post page
POST	/deletepost/{post_id}	Deletes a blog post
GET	/updatepost/{post_id}	Returns the update post page
POST	/updatepost/{post_id}	Updates the blog post information
GET	/viewpost/{post_id}	Returns the single view post page
GET	/viewposts	Returns the viewposts page

Architecture and Features

The below diagram shows the Architecture of the Blog Application:



This blogging application offers a comprehensive set of features for optimal user experience. Users can create, update, and delete their own profiles with ease and Users can also view their profile stats. The search and follow function allow users to discover and keep up to date with their favourite bloggers, and the option to unfollow at any time provides flexibility. The application also supports the full range of CRUD operations for blog posts, enabling users to create, view, update, and delete posts with precision and efficiency as well as for the Users. Overall, our application is designed to provide a professional, streamlined experience for all users.

Video

Video Link → https://drive.google.com/file/d/1YErSI26DqdWbf6u4jw1qU1YSKhDwF_20/view?usp=share_link