

BLOG LITE APPLICATION – PROJECT REPORT

Author

Name : Puppala Bala Sai Manikanta Sandeep

Roll Number : 21f1000485

Student Email ID: 21f1000485@student.onlinedegree.iitm.ac.in

About me : I am studying Btech 3rd Year in Computer Science and Engineering in GITAM University. I am very much passionate about my studies because each time when I do something related to studies or computers I find them very interesting. I always want to sit in front of laptop and do some useful work. I love teaching and motivating others. I want to do team work which I hadn't tried before. And my future goal is to learn and master in Data Science. And coming to this project I have done it in just 17 days in which I took 3-5 days for planning, designing the database and html templates. And I have done the coding and debugging part in rest of the days.

Description

The user of this application should give some information about them in order to onboard the platform. The main objective of this application is to display the data that one person posted to others based on the following criteria. And also managing the data that has been posted should also be needed.

Technologies used

- FLASK - It's a Framework used to create and design web applications easily with python
- JINJA2 - It's used to render HTML pages dynamically by using values, conditions and loops.
- Python - Python is used as backend language for processing requests and data.
- SMTPLIB – It is used for emailing purpose for better login system.
- HTML, CSS, JS - These are used for frontend of the application
- Bootstrap 5.0 – It is used for responsiveness of the application and also for good looking interface.

DB Schema Design

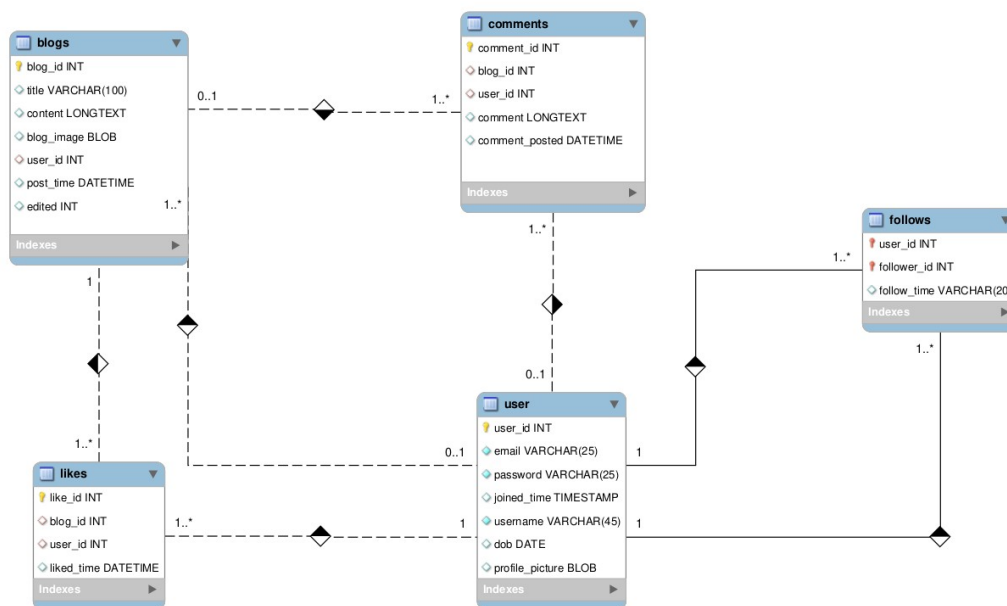


Fig: EER Diagram of Blog Lite Application

For this application, I have designed 5 tables in SQLITE 3, they are user, blogs, likes, comments and follows. The user should enter his/her information to onboard the platform. For that email ID, username and Password is mandatory to login. We can store the likes and comments in a single table but I have separated them because I want to record the timestamp of like. Follows table consist of user id and follower id in which they both form the composite key. Here 1 follows 2 is not same as 2 follows 1 and the composite key preserves this constraint. For creation of a blog there must be a user account, so user_id is foreign key for blogs table. For liking or commenting on a post/blog there must be a blog so, blog id is foreign key for likes and comments table.

Architecture and Features

Bloglite is the name of the root folder. In the Bloglite, there are four sub folders and a main.py file. The four sub folders are templates, application, static and database. In the templates folder there are four files which contains the html templates. In application folder, there are python files including configuration, controllers and some other files related to blog functions. Each file is created for a special purpose of same kind. That is all the functions related to posts will reside under blogfunctions.py, functions related to user reside under userfunctions.py, functions related to follow table reside under followfunctions.py, functions related to comments, likes reside under commentfunctions.py and likefunctions.py respectively. CSS, JS and Images reside under static folder. And the SQLITE3 database reside under database folder in the root directory.

In this application, I have created a login page which will ask for email and password. I have also give links to reset the password and create a new account. For both of these options first the user needs to validate their email, because an OTP has sent to their mails and they need to enter the correct OTP. This was done by using SMTPLIB. After login into account, the user can see some top feeds that are posted by the users those are already there. There is an option to start a post. The user can also search for a post with the title. User can post a new blog with start a post button. The post contains title, content and image where content and image are optional. The user can see his/her profile by clicking on account button. User can update their profile by clicking on edit details option. User can search for an user in the navigation tab using either username or full name. User can edit the post using edit option located in 3 dot button. User can like or comment on the post. User can see his/her posts by clicking on my posts option located in navigation bar.

This application is designed in such a way that it can be used with mobile also. Because of the responsiveness of this application it can be accessed with the help of any device. The design and colour of the application is made to make the user comfortable. The eyepleasing nature of the application makes the user to explore more. Bootstrap 5.0 helped me alot in creating responsive web application. To create the posts I have used Bootstrap 5.0 modals. Modals helped me a lot for seeing the full screen image, editing the post and also for creating the post. The user can create the account, while seeing their profile the user can read the data, they can update the data and also delete the user account. In the same manner, the user can create a blog/post, update the post, read the blogs of themselves or others and also delete their post which makes CRUD on user and blogs.

Implementation of these features are done by the backend python language. For a specific request the request will go to the controllers.py file and there I have used the url in which the essential function gets called from other .py files. The data that is needed is drawn from the database using sql queries and was returned from the function which gets caught by the controllers.py. The data that was received by the controllers.py renders the html template located inside the templates folder using the jinja statements. In this way the entire application is rendered.

Video

Video Link: <https://drive.google.com/file/d/11oknqk6SiYfOhCgCR0Fww1lawb37nyKh/view?usp=sharing>