Bloglite

Name: Abhishek Prasad Prajapati

Roll number: 21f1003546

Email-id: 21f1003546@student.onlinedegree.iitm.ac.in

I am graduate mechanical engineer and currently pursuing this course.

This is a multiuser basic blog website where we can upload our blogs to share with others whom we followed.

Basic functionalities of Bloglite are following:

(1) Feed Lists of blogs uploaded by other users you follow
(2) Blog / Post can be uploaded to share with others which can be edited or deleted by owner of the blog/post.
(3) Search Page can be used to search users using username and follow/unfollow them.
(4) Profile page of users with basic statistics of post count, following/followed by count.

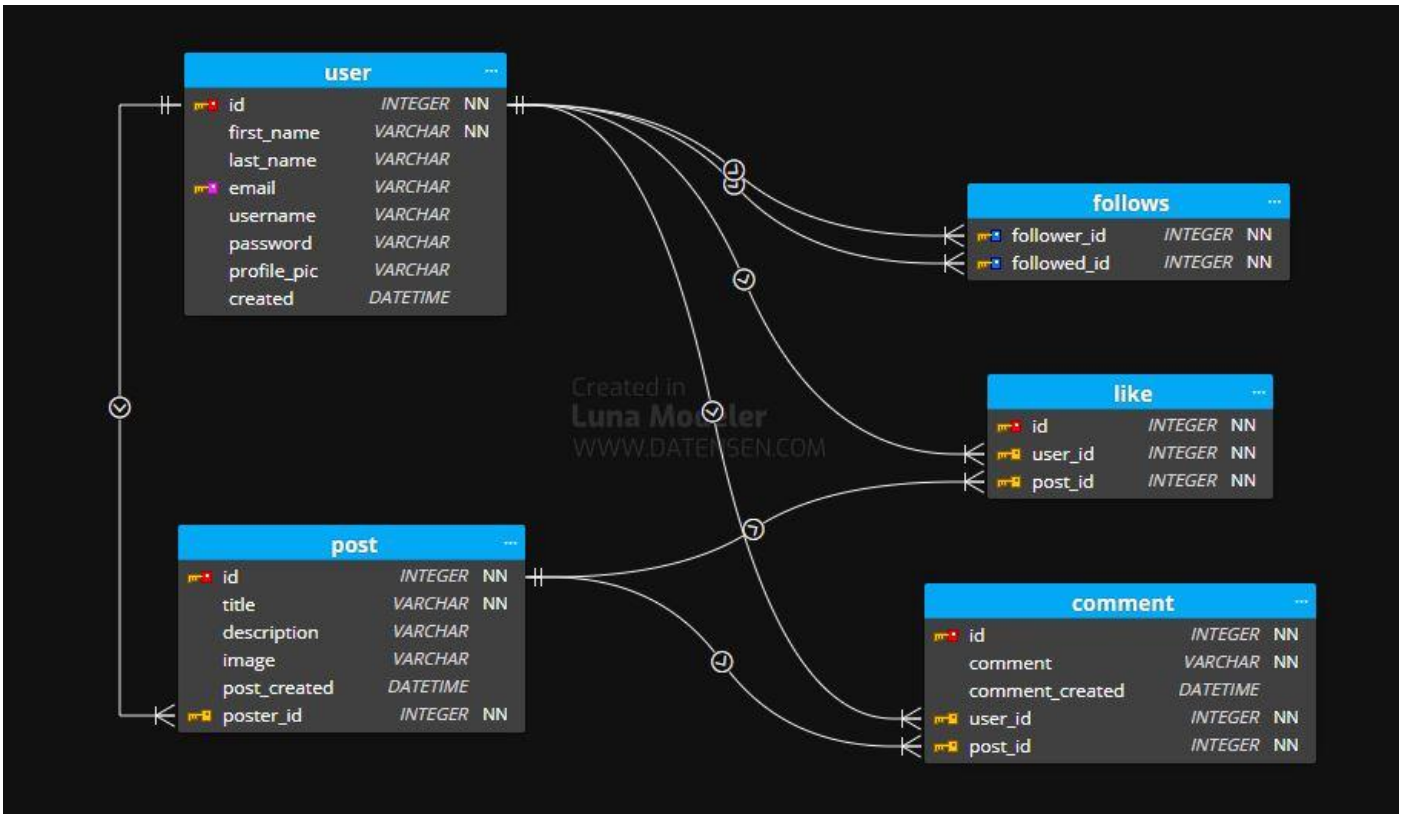
Extra functionalities:

(1) Random suggestions of unknown/unfollowed users on homepage.
(2) Like and Comments on the blog/post with count.
(3) Default image for profile picture if user not upload its profile image.
(4) Default image for blogs if user not upload blog image.
(5) User post information can be exported like username, email, post title, post caption, like counts, comment counts

Technologies used:

- **Application code**: flask, sqlalchemy
- **Database**: sqlite3
- **Style**: Jinja2, CSS, Bootstrap

DB Schema Design:



Tables:

(1) User: user table contains information of user personal details and login authentication details.
(2) Post: post table has information about blogs/posts that user will upload. This table is connected with user table by poster_id having foreign key as user_id.
(3) Follows: follows table is used for many to many relationships between users itself here both follower_id and followed_id have foreign key user_id.
(4) Like and Comment: these table keeps information of likes and comments of post by users which is connected by user and post table by their id.

Architecture and Features:

- **requirements.txt**: This file lists all the Python modules that your app depends on, so that they can be installed using pip install -r requirements.txt.
- **main.py, __init__.py**: This is the main Python file that will run the Flask app. It will import the necessary Flask modules and other libraries, and define the routes for handling requests.
- **blog/views.py, auth.py:** these python files are controllers of app.
- **blog/models.py:** This file contains all information of database tables and relationships.
- **blog/static/**: This folder contains by default images used in app such as background, profile, blog images.
- **blog/templates/**: This folder contains HTML template files that app will render and serve to the client.
- **Blog/static/downloaded_blogs**: This folder contains exported csv files of blogs/post.

Folder Structure

```
.
└── ProjectFolder/
    ├── main.py
    ├── requirements.txt
    ├── README.md
    ├── Project_doc.pdf
    └── blog/
        ├── static/
        │   ├── downloaded_blogs
        │   ├── profile_pics
        │   ├── blog_pics
        │   └── images/
        │       ├── background.jpg
        │       ├── blog-default.png
        │       ├── bloglite.gif
        │       └── default_profile.png
        ├── templates/
        │   ├── 404.html
        │   ├── index_blog.html
        │   ├── login.html
        │   ├── new_post.html
        │   ├── posts.html
        │   ├── other_profile.html
        │   ├── post_restriction.html
        │   ├── post_updated.html
        │   ├── profile.html
        │   ├── register.html
        │   ├── search.html
        │   └── user_update.html
        ├── __init__.py
        ├── auth.py
        ├── models.py
        └── views.py
```

Project Video Link : https://drive.google.com/file/d/1cyLiFNLLNvGc0S1c-cAVHA0p_rif70Ny/view?usp=sharing