## Author

Full Name: Aarya Chetan Motiwala
Roll Number: 21f1003998
Email: 21f1003998@ds.study.iitm.ac.in

## About Me

I am an aspiring Data Scientist, currently in 3rd year of MIT- WPU college as well as in the diploma level of IITM BS degree.I am learning subjects like Statistics, Mathematics, Machine Learning and Application development so that I can create stories from the data.

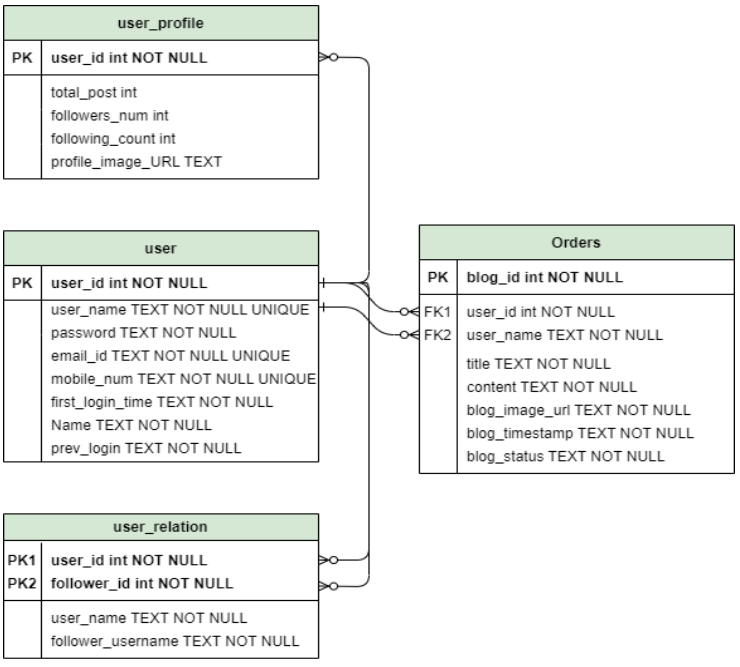
## Description

In this project we had to create a multi-user app designed for uploading blogs with images. Users can upload, edit, delete the blogs multiple times and also every user can search for other users. In this app each user will have their own feed and the system will automatically display the blogs from the users that they follow.

## Technologies used

**Flask**: A micro web framework for building web applications in Python.
**Flask-Cors**: An extension for Flask that provides cross-origin resource sharing (CORS) support.
**Flask-SQLAlchemy**: An extension for Flask that integrates SQLAlchemy, a popular SQL toolkit, into Flask applications.
**Flask-RESTful**: An extension for Flask that simplifies the creation of RESTful APIs.
**flask_bcrypt**: An extension for Flask that provides bcrypt hashing utilities for secure password storage.
**JWT Tokenization**: A technique for securely transmitting data as JSON web tokens (JWTs) between parties in web applications.
**VueJs**: A progressive JavaScript framework used for building user interfaces and single-page applications.
**Redis**: An in-memory data structure store used as a database, cache, and message broker in web applications.
**Celery**: A distributed task queue used for background job processing and scheduling in web applications.

## DB Schema Design



## API Design

BlogLite is a multi-user app which is used for uploading and viewing  blogs with images.
My api's four primary categories were User, Profile, Blog, and Feed. User primarily comprises user-related operations like login and registration. Then, Profile consists of operations related to user profile like following, followers, profile and search. Then Blog consists Operations related to blogs like uploading, updating , deleting and viewing the blogs. Then the feed category consists  the operations that would happen on loading of feed page that is automatically display the blogs of the users that they follow

## Architecture and Features

In my submission file there are 2 main folder one is backend and other one is frontend the backend folder consists of 4 major folder and 2 files .
First one is application folder which consist of

1. config.py-(apps configuration)
2. cache.py-(caching setup)
3. database.py-(database linking with SQLAlchemy
4. Models.py-(models)
5. operations.py-(api implementation / controller)
6. workers.py-(used for celery)
7. token_validation.py( for jwt token_validation in operations.py)

Second is db_directory

1. bloglitenew.sqlite3-(database)

Third is static folder which consists static file like logo
Fourth is templates which consists templates for daily and monthly reminder
main.py-(run the backend app and celery implementation)
requirements.txt-(modules to install before running the app)
frontend folder consists of index file which is linked to all the other components
Then router.js is where the vue app is initialised and the routes are mentioned
Then the components folder consists of all the vue components like login, search, feed etc.

Features

1. User signup and login
2. User profile view with basic stats
3. Blog Post Management
4. Search and Follow / Unfollow Others
5. User’s Feed
6. Daily Notification
7. Monthly Notification

## Video :  [Project video link]

https://drive.google.com/file/d/1TwhOGcc9QGH5RAj2WEl1MAhiqLX_pyAU/view?usp=share_link