ECHO

CS253 Project Presentation

BitByBit

GitHub:- https://github.com/aryank2240/CS253-BitByBit-Project

INTRODUCTION

The product, Echo aims to provide a platform for all those in campus who are interested to either write or read. Whether you are interested in sharing your thoughts about something new you learnt, raise your voice against some injustice or may be just want to ask general questions to the community or just simply want to read other's writing, this platform is for you.

WHY ECHO?

Have you ever felt the need to share your thoughts to the campus and wanted to know what others think about that or maybe wanted to ask a question to the whole community or wanted to read yours peers' writing? Yes! Well, Echo gets you covered.
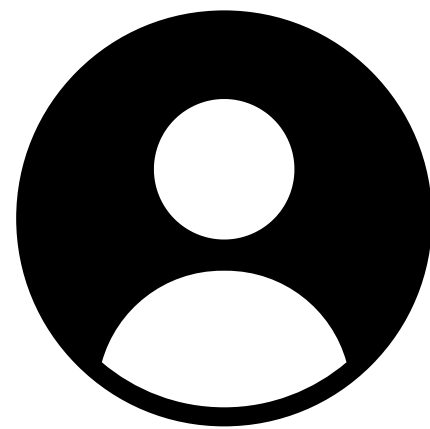
Intented Audience

Currently, the platform is open only for the IITK community. Users can be either a writer or a reader.
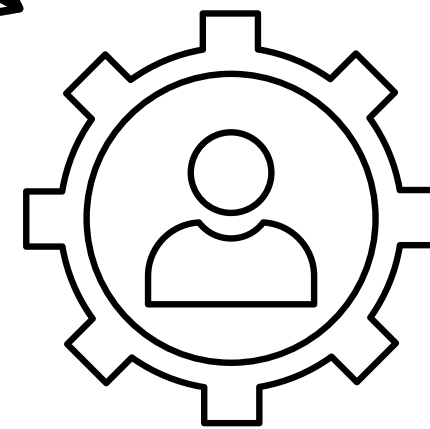
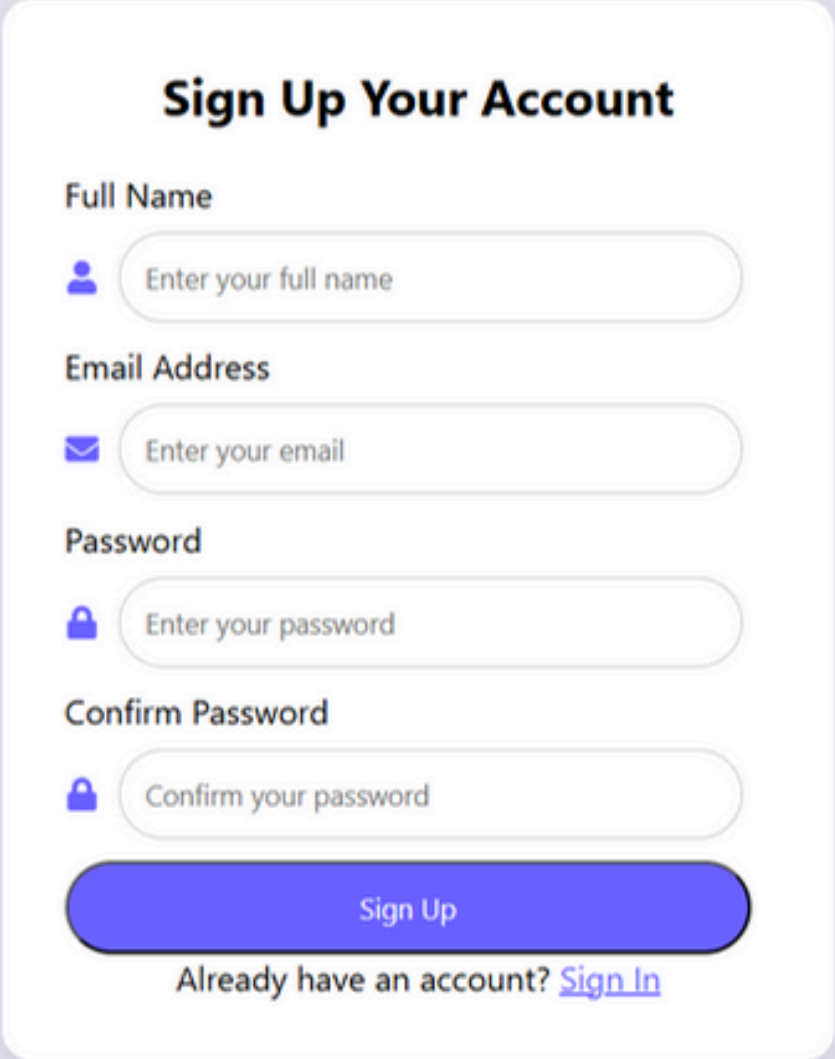# NOW LETS TALK ABOUT WHAT OUR SOFTWARE CAN DO?
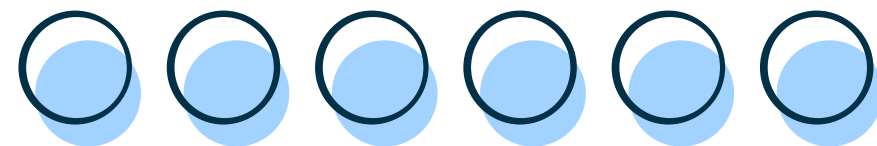
Accessing the software as

User

Admin

Accessing the software as a User.

- **CREATE ACCOUNT: As a user you can create a verified account and use it to access the website.**

Accessing the software as a User.

- **PASSWORD RECOVERY: If the user forgets his/her password then he/she can easily recover the password using the forgot password option.**

Accessing the software as a User.
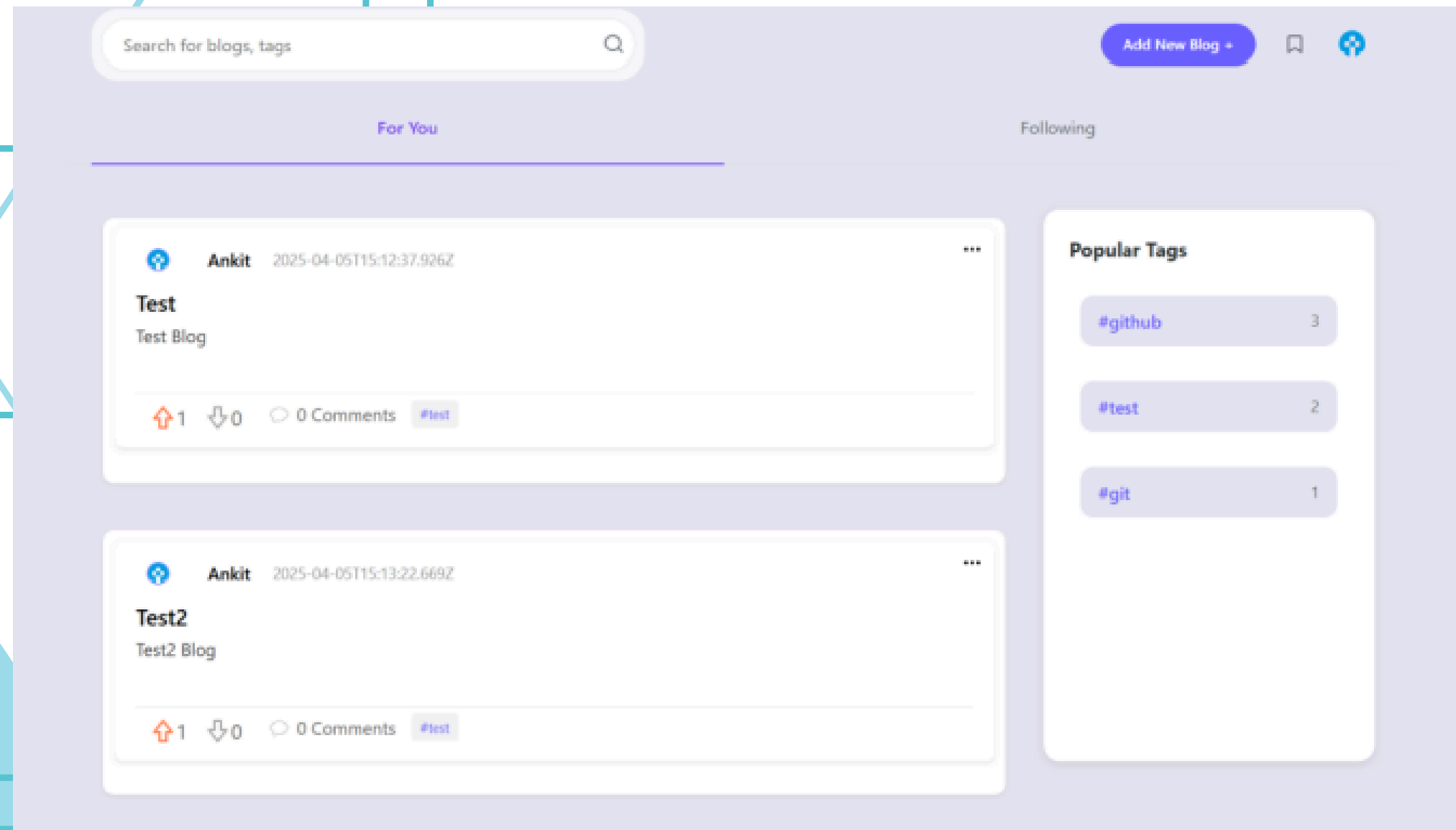
Home page

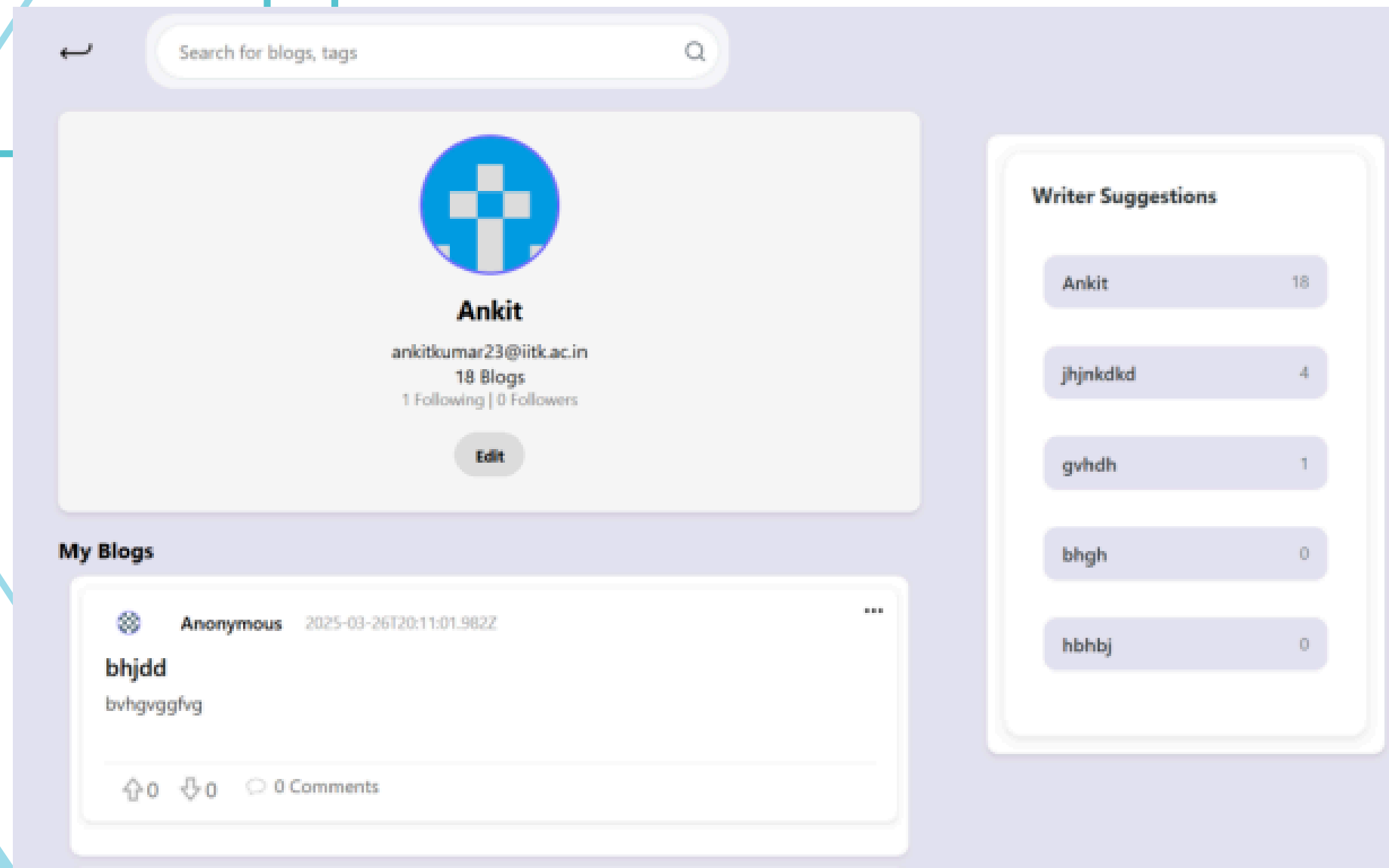This is the home page of our software. The software was developed keeping in mind that it should be user friendly and easy to use. Users can further perform multiple tasks on the website that are :

Accessing the software as a User.

Profile page

Users can use this page to edit their profile picture and their names which would be displayed under their blogs. They can also view their previously published blogs on this page.

Accessing the software as a User.

- **CREATE BLOG:** User can easily write and publish a blog using his account.

- **ADD TAGS:** Different tags can be added to categorize the blogs

- **PUBLISH:** Blogs can be published either publicly or anonymously which means your id will not be visible to others.
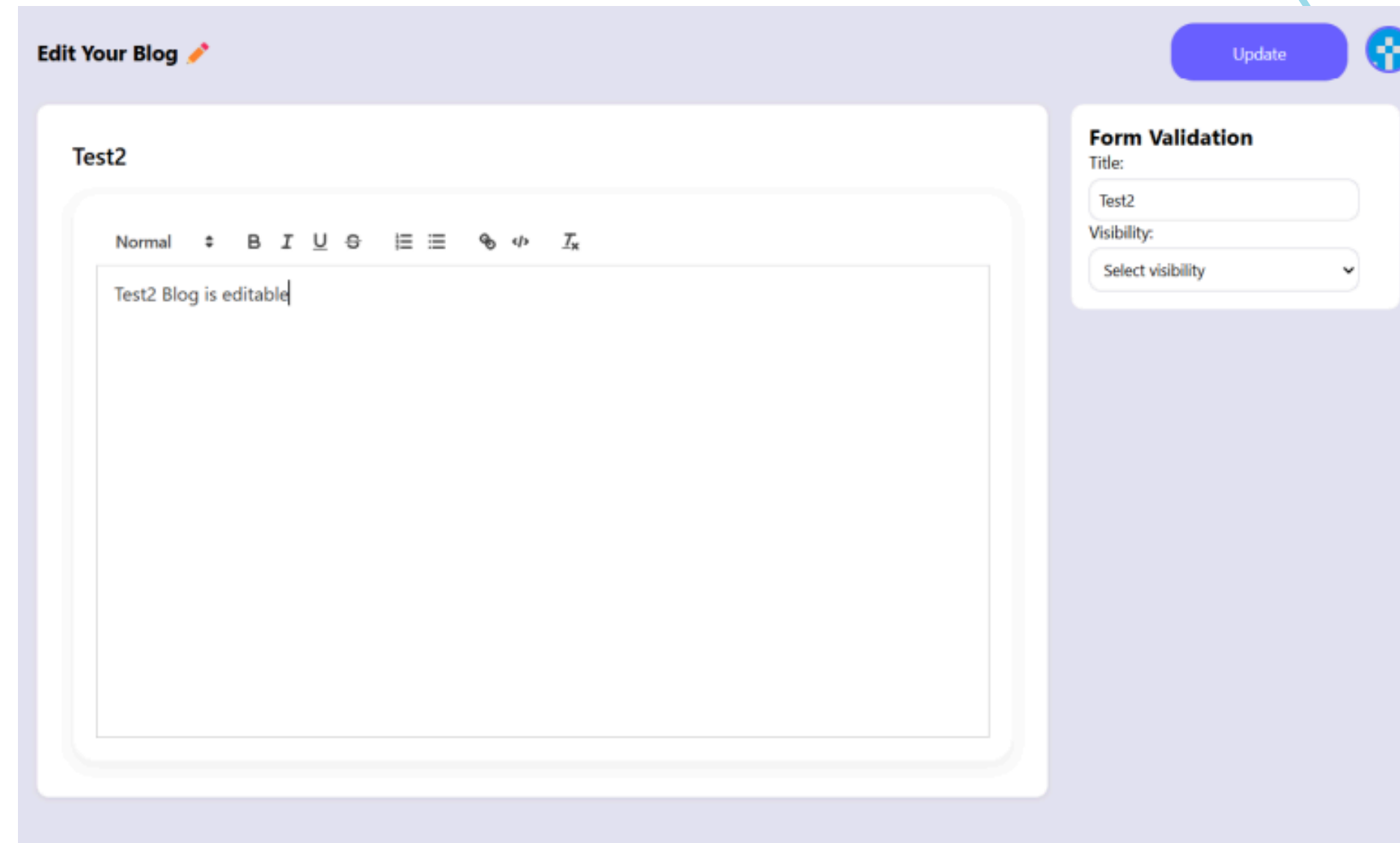

Create blog page

Accessing the software as a User.

- **EDIT BLOG: If you made a mistake or want to change something in a blog then no problem! You can easily edit your previously uploaded blogs using the edit blog feature.**


Edit blog page

Accessing the software as a User.

- **FOLLOW: The website does not only target the writers but also the readers who can use the follow function to follow other users to keep up with their work.**

- **UPVOTE/DOWNVOTE: User can upvote or downvote other's work which helps in maintaining a feed in which the user might be interested.**

- **COMMENT: User can comment on someone else's work using the comment option.**
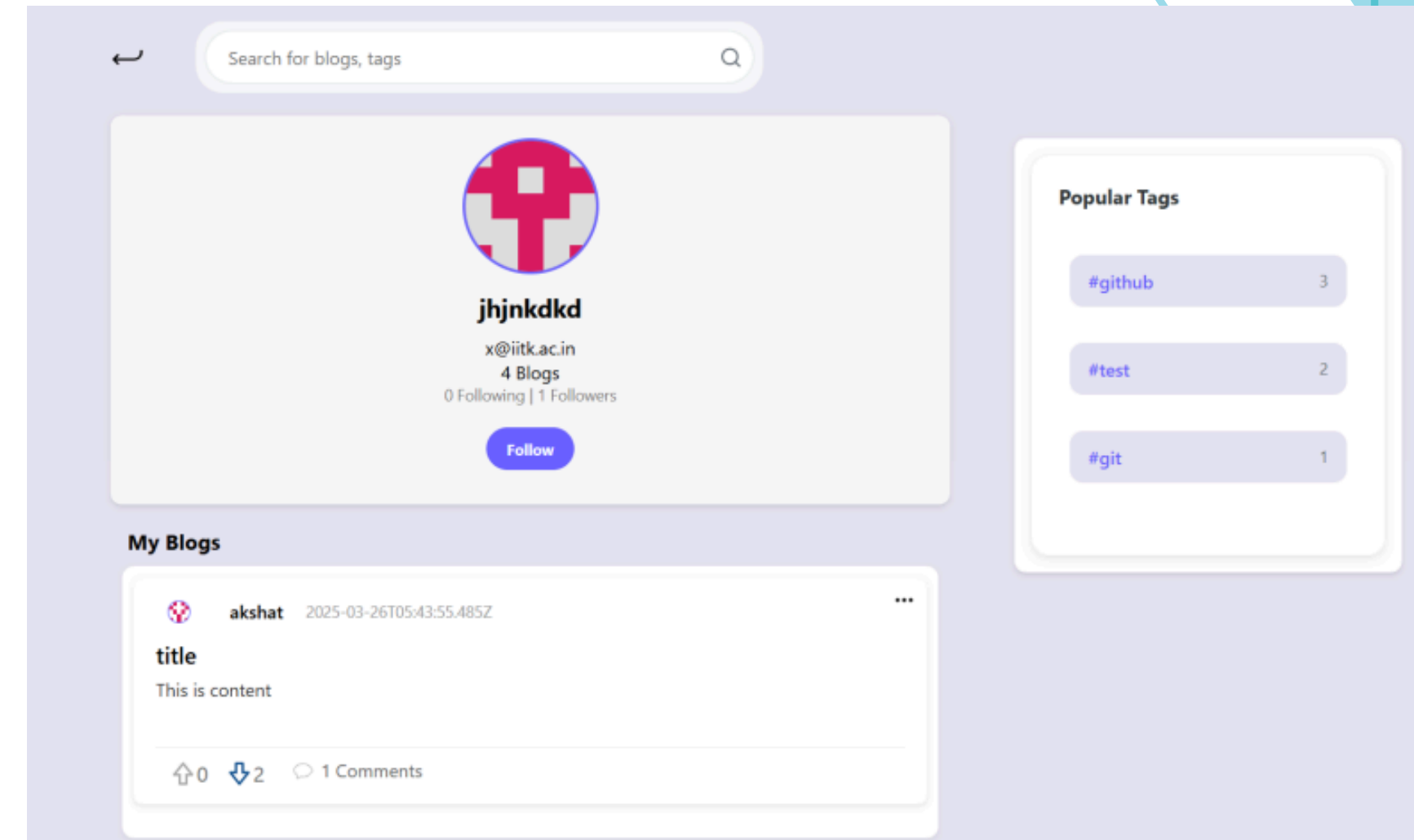
Follow page

Accessing the software as a User.

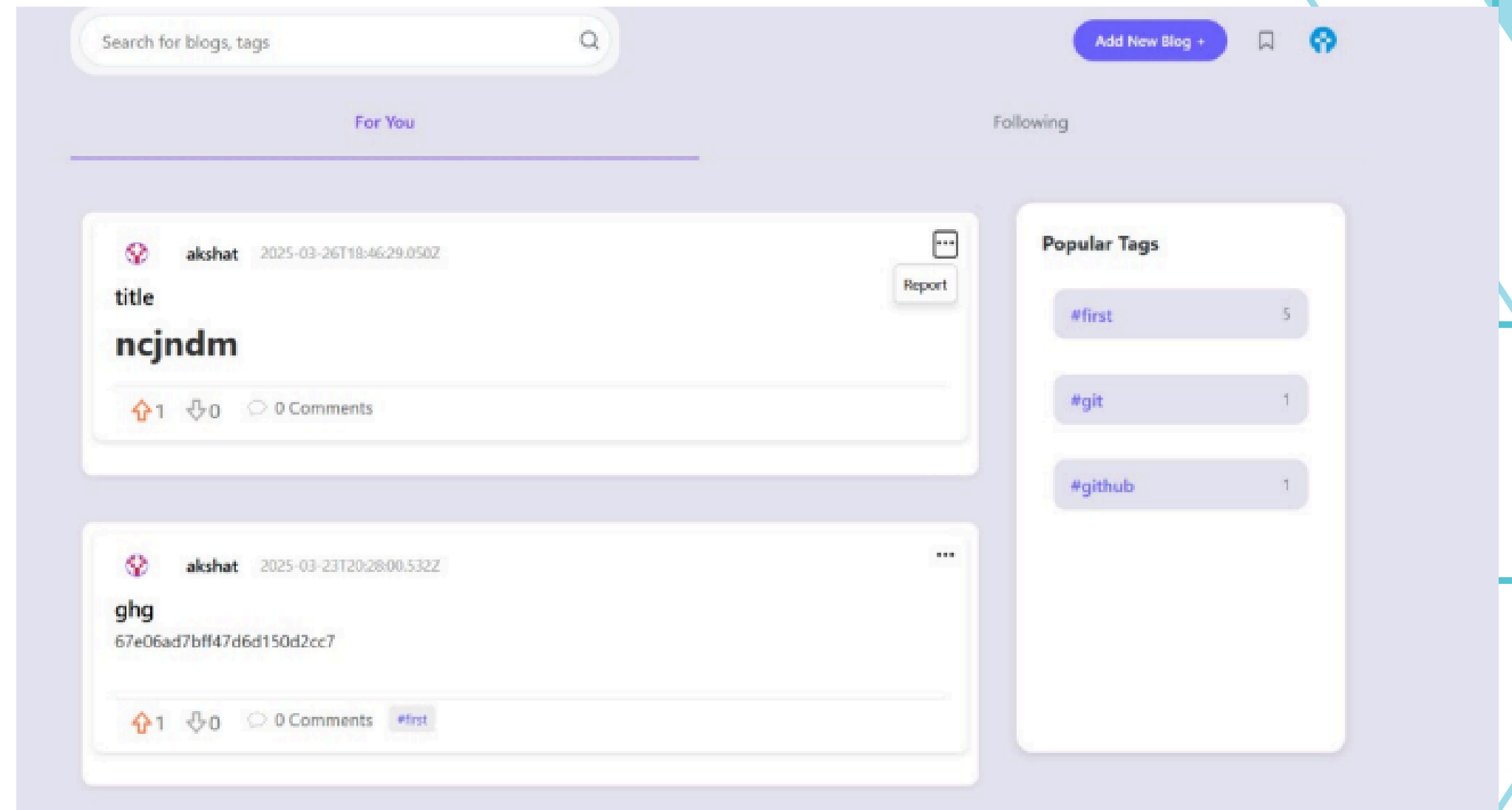- **Users can also delete or edit their comments they made on someone else's posts.**
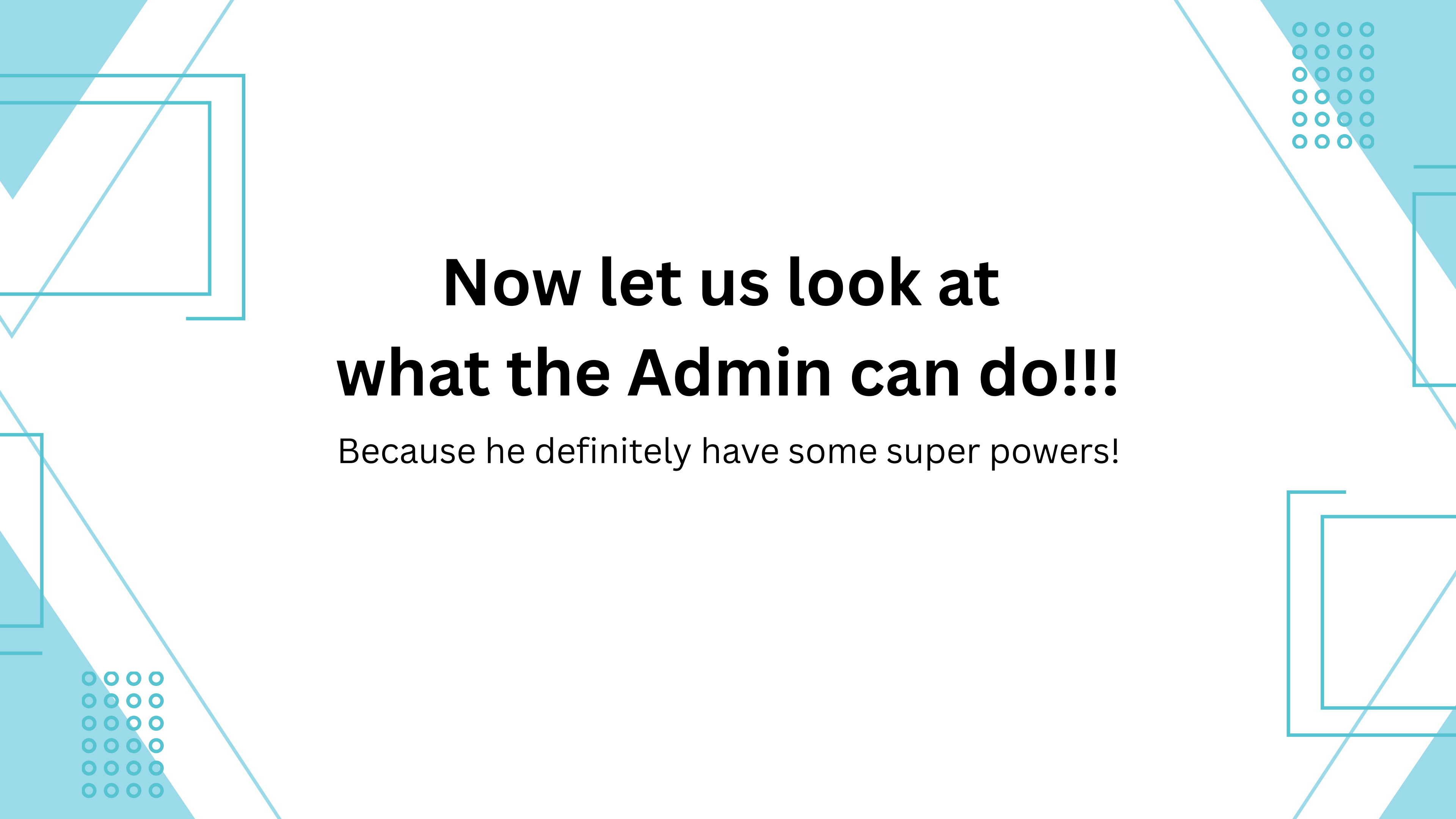
Accessing the software as a User.

- **REPORT: If a user finds something offensive and thinks that the text is not people friendly then the user can report the blog. The reported blog will be verified by the admin. The admin can take actions on the reported blog like deleting the blog.**

Accessing the software as a User.

- **LOGOUT: Obviously if you are fed up with the software or just want to take a break, we have also added a log out button for you**
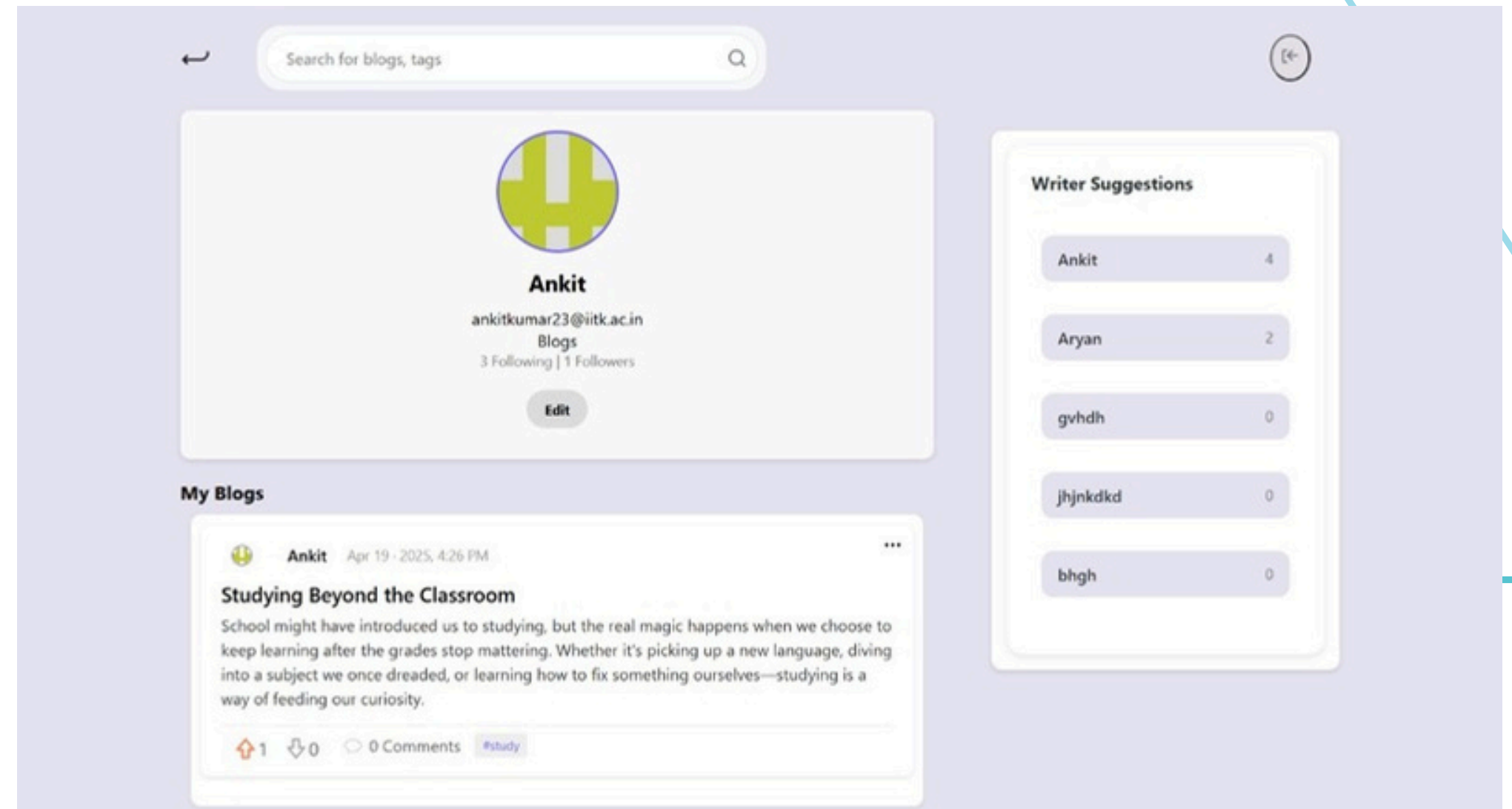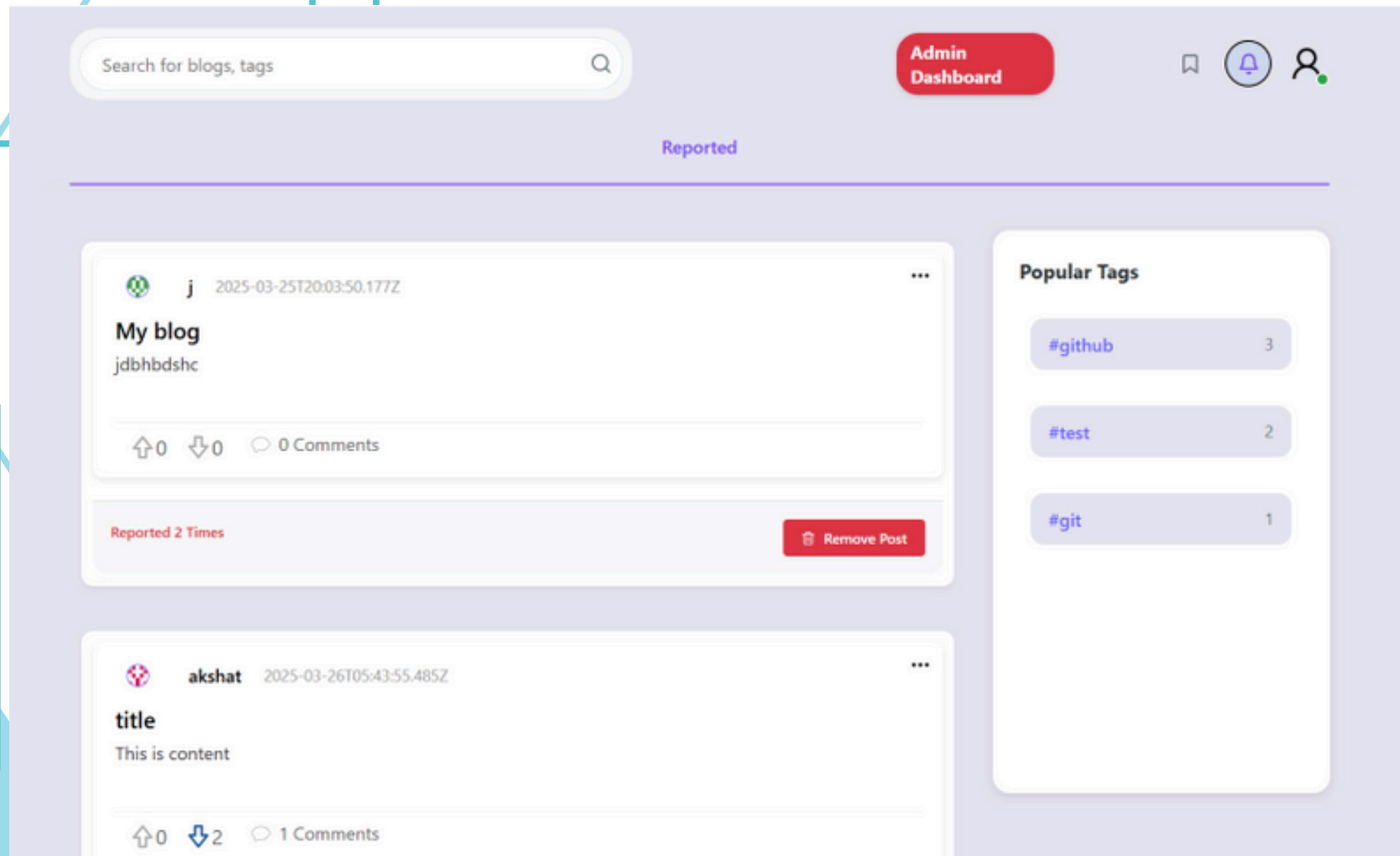
Now let us look at what the Admin can do!!!

Because he definitely have some super powers!
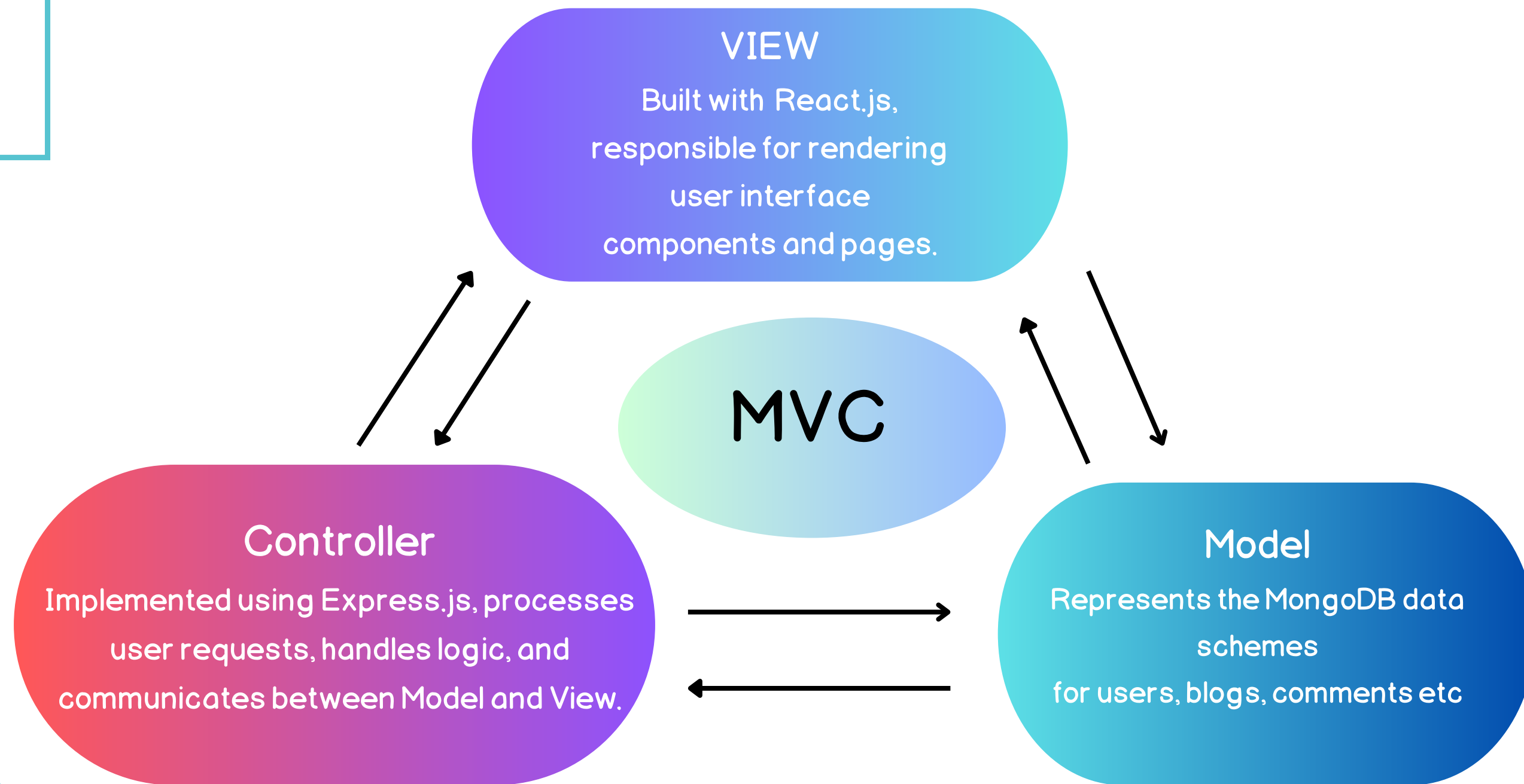
Admin Home page

Well the admin can do whatever a user can do, such as blog writing, reading, editing so we will skip that part and look the the superpowers of admin.
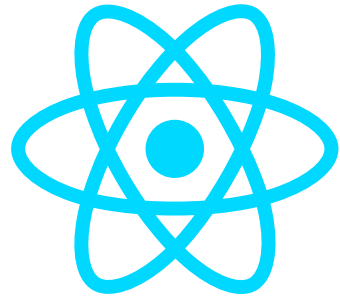
The Admin can delete anyone's post.

IMPLEMENTATION

- **Architecture**
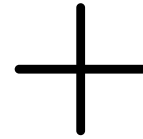- **Frontend**
- **Backend**
- **Database**

MODEL ARCHITECTURE

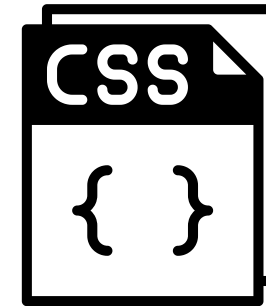**VIEW**
Built with React.js, responsible for rendering user interface components and pages.

MVC

Controller
Implemented using Express.js, processes user requests, handles logic, and communicates between Model and View.

Model
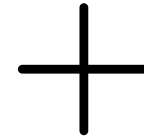Represents the MongoDB data schemes for users, blogs, comments etc

FRONTEND

React.js

HTML

CSS

Modular, component-base
UI framework for fast rendering.

Defines the structure of
each page and its content.

Handles layout, design,
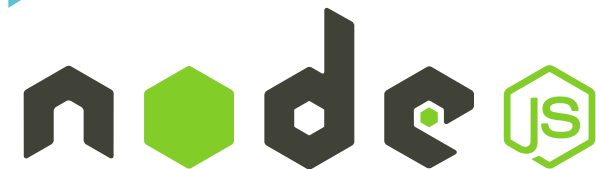responsiveness and virtual consistency.

📦 /frontend

├ 📁 components/ → Reusable UI blocks
(Navbar, Cards, etc.)
├ 📁 pages/ → Views like Home, Profile,
Blog Editor
├ 📁 services/ → API calls and business
logic
└ 📁 public/ → Static files (index.html, icons,
manifest)

BACKEND

node JS
JavaScript runtime for building fast,
scalable server-side applications.

Express JS
Minimal and flexible framework
for creating RESTful APIs.

/backend

├ routes/ → API endpoints (auth, blogs, users)
├ controllers/ → Business logic for handling requests
├ middleware/ → Auth & validation layers
├ models/ → MongoDB schemas for Users, Blogs, Comments
└ utils/ → Helper functions (OTP, formatters, etc.)

DATABASE
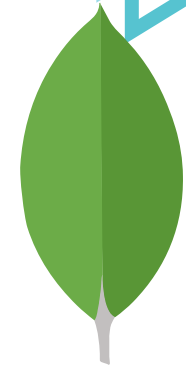
MongoDB

A NoSQL, document-oriented database designed for flexibility and scalibility.

📦 /Database (MongoDB)

👤 Users   → User profiles, login info, settings .
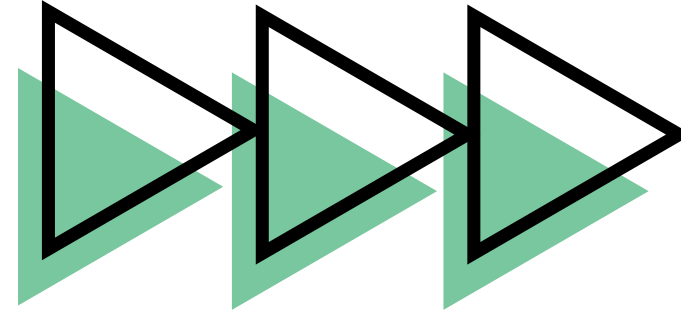
📝 Blogs   → Blog posts (title, content, author, tags)

💬 Comments   → Comments on blogs.

👥 Followers   → Following/follower relationships.

DEMONSTRATION ▷▷▷

LESSONS LEARNT

- Gained hands-on experience building dynamic, interactive web pages for real-world applications.
- Learned to organize code into modular sections and use GitHub for seamless collaboration, streamlining maintenance.
- Met project deadlines through structured planning and teamwork, strengthening our time-management and collaboration skills.
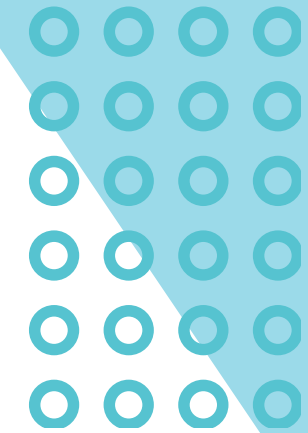
FUTURE
DEVELOPMENT PLANS

PLANS

- Collaborative Blogging
- Themes and Templates
- Embedded Content
- Direct Messaging
- Polls and Surveys
- Better UI Design

Our Team

Course Instructor : Dr. Indranil Saha

Project Mentor : Paras Ghodeshwar

Group members:

- Ansh Adarsh
- Aryan Kumar
- Durbasmriti Saha
- Gone Nishant
- Govind Nayak Jarabola

- Harsh Bhati
- Lavish Kanwa
- Lokesh Kumar
- Someshwar Singh

THANK YOU