**0. Overview:**

**Project Name:** blogy

**Author:** Karam , Anas

**Date:** 22\3\2025

**Objective:**

A blog app where users can upload posts, imgs, likes , dislikes and comments others posts.

## **1. Technologies:**

**1.1. Front-end:** Html, Css, Javascript

**1.2. Back-end:** Node.js with express.js

**1.3. Database:** MongoDB with Mongoose

**2. End-Points:**

2.1 [POST]/login

2.2 [POST]/register

2.16 [POST] /logout

2.3 [GET]/posts

// 2.4 [GET]/posts/:postId/comments

2.13 [POST]/posts/:postId/comments

2.5 [GET]/user/posts

2.6 [POST]/user/posts

2.7 [PUT]/user/posts/:postId

2.8 [DELETE]/user/posts/:postID

2.9 [GET]/admin/users

2.14 [PUT] /admin/users/:userId/block

2.15 [PUT] /admin/users/:userId/unblock

2.10 [GET]/admin/:userId/posts

2.12 [DELETE] /admin/posts/:postId

**3. Modules:**

**3.1 User:**

Can (Login , Register , Create Posts , like Posts, Dislike , comment , Delete Posts , Updated Posts ,See Posts)

**3.2 User (Admin):**

Can (Login , See all users , See all posts , BLock users , Delete posts ,comment)

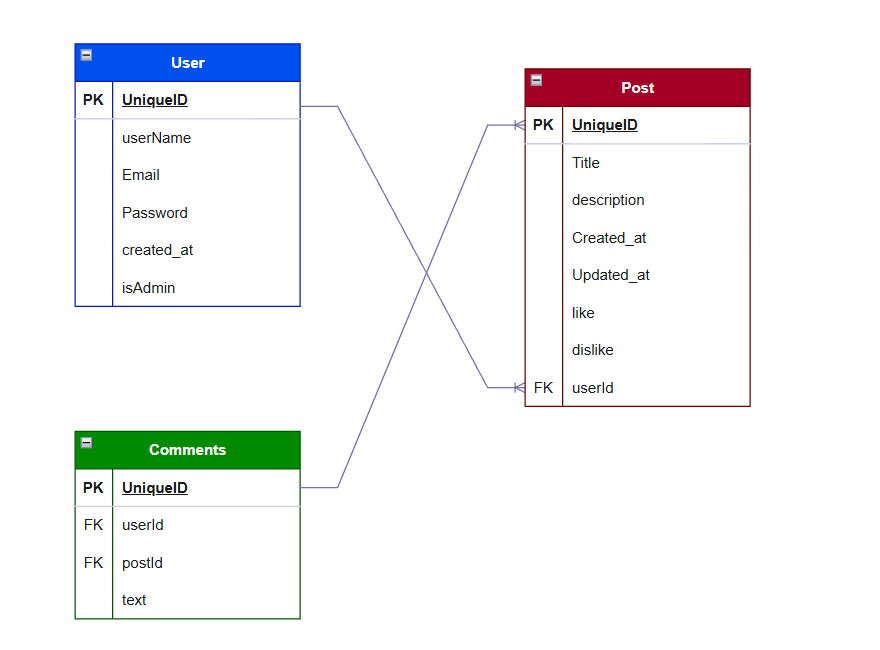
**3.3 Posts:**

Has like and dislike (Counter)

**3.4 Comments:**

Users can comments on posts

**4. DataBase:**

\*\* created\_at 

**5. Files structure:**

**Blogy:**

– Front-End

–HTML

–CSS

–JS

– Back-end

– App

– Controllers

– routs

– Database

**6. Security:**

6.1 Authentication using express-session

6.2 Password hashing using bcrypt with salt

6.3 Session store in Memory or MongoDB

6.4 Authorization Custom middleware

// 6.5 Input validation express validator or manual

// 6.6 Express Rate limit

// 6.7 Csrf

// 6.8 MongoDB Injection Prevention