## Social Media App — Full Documentation

## **\*** Tech Stack

Layer	Technology
Frontend	Angular
Backend	Express.js (Node.js)
Database	MongoDB (Mongoose)
Storage	Cloudinary (image uploads)
Auth	JWT (JSON Web Token)
Notificatio	Nodemailer (Email)

### Features

n

- V User Register/Login with JWT
- **V** Create posts with text + image (Cloudinary)
- **V** Like and **unlike** posts
- Add and delete comments on posts
- V Follow/Unfollow users
- V Feed to show posts from followed users
- **V** Delete your own posts
- **Search users** by name or email
- V Email notification when someone you follow posts

## **Project Structure**

### Backend Details

## **Models**

#### User

```
username: String,
email: String,
password: String,
followers: [UserId],
following: [UserId],
notifications: [String]
```

#### **Post**

```
js
CopyEdit
{
 author: UserId,
 content: String,
 imageUrl: String,
```

```
likes: [UserId],
comments: [{
user: UserId,
text: String,
createdAt: Date
}],
createdAt: Date
```

#### Auth

- JWT issued on login.
- Sent in Authorization: Bearer <token> header.
- Middleware auth.middleware.js protects all secured endpoints.

## Cloudinary Integration

- Upload images from Angular.
- Backend uploads to Cloudinary using cloudinary.v2.
- Image URL is saved in MongoDB with the post.

## **Email Notifications**

- Sent to followers when a user creates a post.
- Handled with nodemailer.

## **Solution** Backend API Endpoints

Method	Endpoint	Description
POST	/api/auth/register	Register a new user
POST	/api/auth/login	Login and receive JWT
GET	/api/posts/feed	Get posts from followed users
POST	/api/posts	Create a new post
PUT	/api/posts/:id/like	Like a post
PUT	/api/posts/:id/unlike	Unlike a post
DELET E	/api/posts/:id	Delete a post
POST	/api/posts/:id/comments	Add comment to post
DELET E	/api/posts/:postId/comments/:id	Delete a specific comment
POST	/api/users/:id/follow	Follow a user
POST	/api/users/:id/unfollow	Unfollow a user
GET	/api/users/search?q=keyword	Search users by name or email

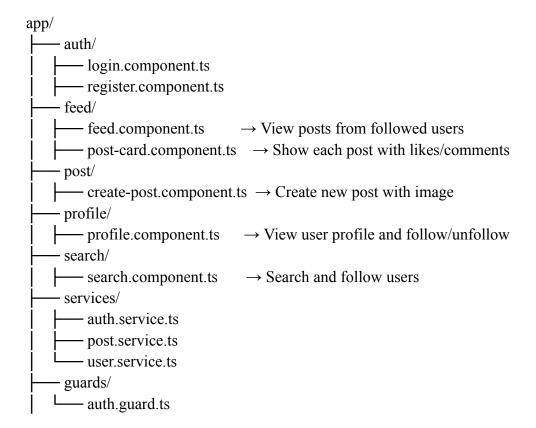
## **(Angular)** Frontend (Angular)

## **Route Guards (auth.guard.ts)**

- Protects routes like /feed, /create-post, /profile, etc.
- If not authenticated, redirects to /login.

# \* Angular Components

pgsql CopyEdit



## Frontend Features Summary

Feature	Component
Register/Login	auth/
View Feed	feed/feed.component.ts
Like/Unlike Post	post-card.component.ts
Comment on Post	post-card.component.ts
Create Post	post/create-post.component.ts
Delete Post	post-card.component.ts (only owner)
Follow/Unfollow Users	profile.component.ts
View & Search Users	search.component.ts

## **Environment Variables**

#### Backend .env:

MONGO\_URI=mongodb://localhost:27017/linkedin-clone CLOUD\_NAME=dyt2h3pgq CLOUD\_API\_KEY=387575578193138 CLOUD\_API\_SECRET=3m\_ITFUhKqMuCYx4s0g3pi1AGxw EMAIL\_USER=your\_email@gmail.com EMAIL\_PASS=your\_email\_password\_or\_app\_password

### **Future Enhancements**

- Real-time notifications
- Bookmark/save posts
- Infinite scroll
- Activity history
- Profile image support
- Full messaging system