

▼ backend

▼ models

JS Post.js

> node\_modules

▼ routes

JS posts.js

JS users.js

{ } package-lock.json

{ } package.json

JS server.js

▼ my\_day1

> angular

backend > JS server.js > app.listen() callback

```
1  const express = require('express');
2  const mongoose = require('mongoose');
3  const cors = require('cors');
4
5  const postsRouter = require('./routes/posts');
6  const usersRouter = require('./routes/users');
7
8  const app = express();
9
10
11  app.use(cors({
12
13    origin: 'http://localhost:4200',
14    methods: ['GET', 'POST', 'PUT', 'DELETE', 'PATCH'],
15    allowedHeaders: ['Content-Type', 'Authorization']
16  }));
17
18  app.use(express.json());
19
20
21  app.use('/api/posts', postsRouter);
22  app.use('/api/users', usersRouter);
23
24  mongoose.connect(
25    'mongodb+srv://elsayedmohamed01065813602:drUimful719CFgwZ@cluster0.wzvepiy.mongodb.net/nti?retryWrites=true&w=majority&appName=Cluster0'
26  )
27  .then(() => console.log(" MongoDB Connected"))
28  .catch((err) => console.error(" MongoDB connection error:", err));
29
30  const PORT = process.env.PORT || 3000;
31  app.listen(PORT, () => {
32    console.log(` Server is running on http://localhost:${PORT}`);
33  });
```

```
const express = require('express');
const router = express.Router();
const Post = require('../models/Post');

router.get('/', async (req, res) => {
  const posts = await Post.find();
  res.json(posts);
});

router.post('/', async (req, res) => {
  const newPost = new Post(req.body);
  await newPost.save();
  res.json(newPost);
});

router.delete('/:id', async (req, res) => {
  await Post.findByIdAndDelete(req.params.id);
  res.sendStatus(204);
});

router.put('/:id', async (req, res) => {
  const updatedPost = await Post.findByIdAndUpdate(req.params.id, req.body, { new: true });
  res.json(updatedPost);
});

router.get('/:id', async (req, res) => {
  const post = await Post.findById(req.params.id);
  res.json(post);
});

module.exports = router;
```

kend > routes > JS users.js > ...

```
1 const express = require('express');
2 const router = express.Router();
3 const mongoose = require('mongoose');
4 const jwt = require('jsonwebtoken');
5
6
7 const userSchema = new mongoose.Schema({
8   id: Number,
9   name: String,
10  imgurl: String,
11  password: String
12 });
13
14 const User = mongoose.model('User', userSchema);
15
16 router.post('/login', async (req, res) => {
17   const { name, password } = req.body;
18
19   try {
20     const user = await User.findOne({ name, password });
21
22     if (!user) {
23       return res.status(401).json({ message: 'Invalid credentials' });
24     }
25
26     const token = jwt.sign({ id: user.id, name: user.name }, 'secretKey', { expiresIn: '1h' });
27
28     res.json({ token, user });
29   } catch (err) {
30     res.status(500).json({ message: err.message });
31   }
32 });
```

```
PS C:\Users\PC\Desktop\Project Nti\backend> node server.js
```

```
Server is running on http://localhost:3000
```

```
MongoDB Connected
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

NOTE: Raw file sizes do not reflect development server per-request transformations.

→ **Local:** <http://localhost:4200/>

→ press **h + enter** to show help



posts

users

Filter 

Type a query: { field: 'value' }

Reset

Apply

Options ▶

#### QUERY RESULTS: 1-4 OF 4

```
_id: ObjectId('6885b4e5e957f9899b2dd718')
id: 1
name: "El-Sayed Mohammed"
imgurl: "https://westerndeserttravel.com/wp-content/uploads/2024/07/unnamed-4-1..."
password: "123"
```

```
_id: ObjectId('6885b4e5e957f9899b2dd719')
id: 2
name: "Ahmed Mohammed"
imgurl: "https://westerndeserttravel.com/wp-content/uploads/2024/07/1adc14f8-ef..."
password: "1234"
```

```
id: ObjectId('6885b4e5e957f9899b2dd71a')
```

posts

users

Filter 

Type a query: { field: 'value' }

Reset

Apply

Options 

QUERY RESULTS: 1-4 OF 4

```
_id: ObjectId('68868dd2e957f9899b2dddc4')
title: "Angular Basics"
body: "Learning components, bindings, and directives."
userid: 1
date: "2025-07-22T00:00:00Z"
imgurl: "https://www.egypttoursportal.com/images/2025/04/The-Red-Sea-Egypt-Tour..."
like: true
likes: 0
▶ comments: Array (empty)
```



\_id: ObjectId('68868dd2e957f9899b2dddc5')

title: "Angular Basics"

body: "Learning components, bindings, and directives."

userid: 2

date: "2025-07-22T00:00:00Z"





**Username**

El-Sayed Mohammed

**Password**

...

**Login**

MyDay1

+

⌵

⌵

←

→

↺

🔒

📄

http://localhost:4200/posts

☆

👤

⌵

⌵


⌵

⌵

⌵

⌵

El-Sayed Mohammed



El-Sayed Mohammed


Login

Posts


About

Login


Angular Basics




Angular Basics



Angular Basics



Angular Basics



🔍

🔧 Inspector

📄 Console

🔧 Debugger

↕ Network

🛠 Style Editor

🕒 Performance

🧠 Memory

📁 Storage

👤 Accessibility

🧩 Application

🔧

▶ 📁 Cache Storage

▼ 📁 Cookies

▶ 📁 Indexed DB

▼ 📁 Local Storage

▶ 📁 Session Storage

🔍 Filter Items

Key

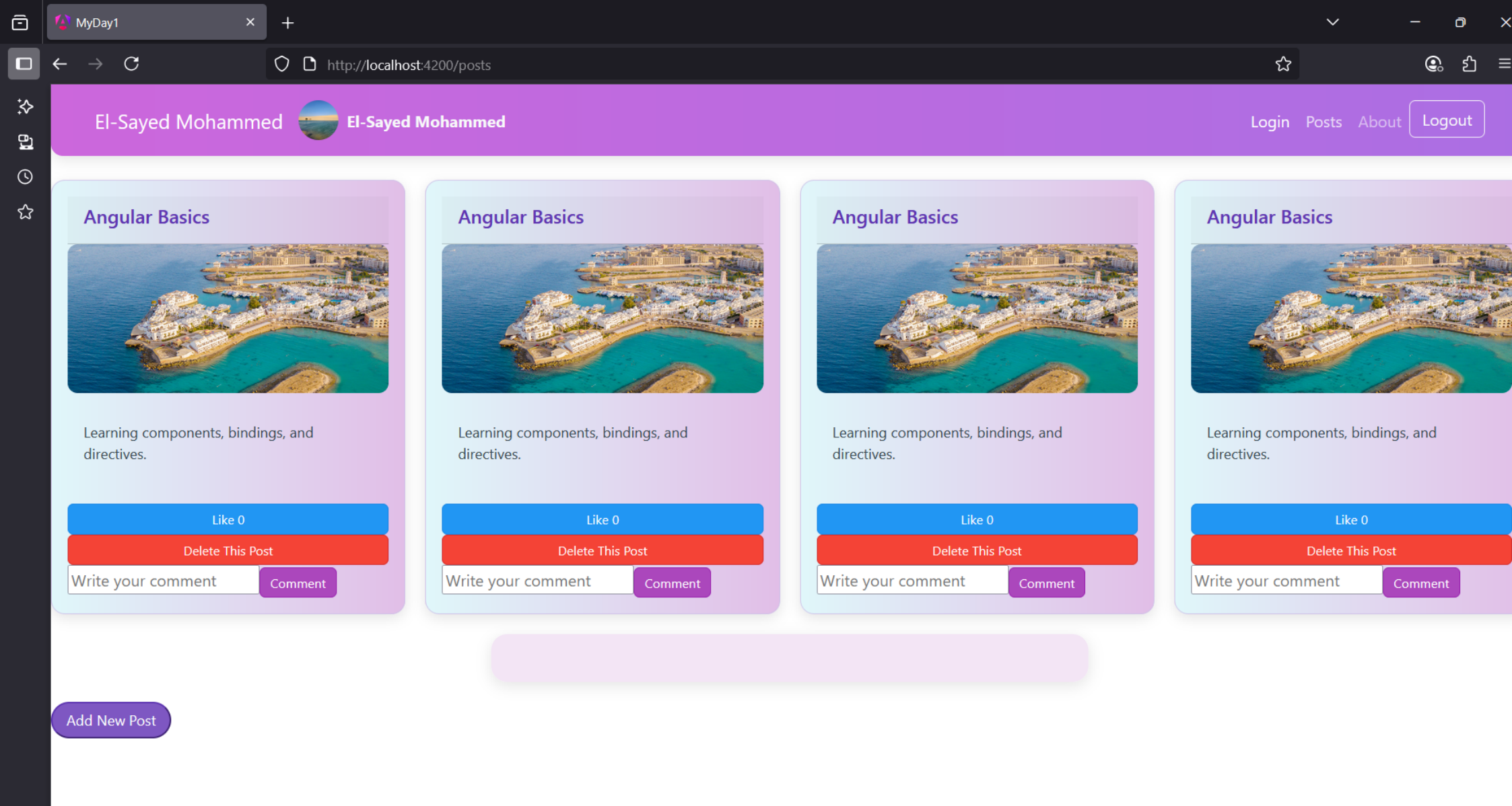
Value

token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwibmFtZSI6IktVbWVkeiE1vaGFtbWVklwiWF0JjoxNzUzODQ4MjI1LCJleHAiOiJlbnR5cS9Gf7xhuDNVesWHvcvY0bcmxWo7Eqs

user

{ "\_id": "6885b4e5e957f9899b2dd718", "id": 1, "name": "El-Sayed Mohammed", "imgurl": "https://westerndeserttravel.com/wp-content/uploads/2024/07/unnamed-4-1.png", "password": "123" }



## Angular Basics



Learning components, bindings, and directives.

Like 0

Delete This Post

Write your comment

Comment

## Angular Basics



Learning components, bindings, and directives.

Like 0

Delete This Post

Write your comment

Comment

## Angular Basics



Learning components, bindings, and directives.

Like 0

Delete This Post

Write your comment

Comment

## Angular Basics



Learning components, bindings, and directives.

Like 0

Delete This Post

Write your comment

Comment

Add New Post



# El-Sayed Mohammed

## Mean Stack Developer

web developer specialized in the MEAN Stack: MongoDB – Express – Angular – Node.js I work on both the front-end and back-end of web applications — from designing responsive and user-friendly interfaces to building robust APIs and managing databases efficiently. I help businesses grow by designing and developing scalable, intuitive web applications. With a strong passion for clean code and smooth user experiences, I empower companies to build digital platforms that boost engagement and deliver measurable results.


---

### Skills:

- ♦ Angular - TypeScript - RxJS
  - ♦ Node.js - Express
- ♦ MongoDB - Mongoose
- ♦ HTML - CSS - Bootstrap

---

### Contact Me:

 **Phone:** +20 1065813602