

Project: Retro Blogs

Tech Stack:

{Reactjs – Frontend

[MongoDB,Expressjs,Nodejs] – Backend}-Mern Stack

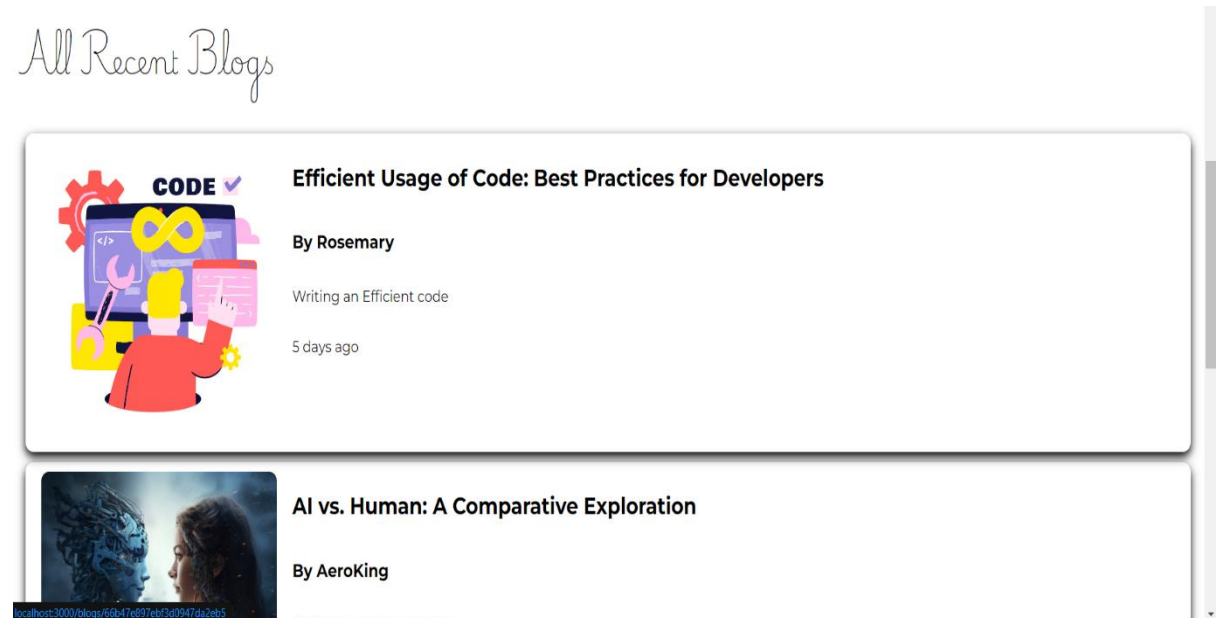
JWT (Json Web Tokens) – Authentication

Routes Unit Test Results:

Blog Route Tests:

1. Test case 1: Fetch all Blogs from the Database **Route :/api/blog/**

Result:



It works perfectly

2. Test case 2: Fetch Single Blog from the Database **route:/api/blog/:id**

Result

Efficient Usage of Code: Best Practices for Developers

Author : Rosemary

Created At : 5 days ago



About : Writing an Efficient code

Introduction

Writing efficient code is essential for building robust, scalable, and maintainable software. Efficient code not only enhances performance but also improves readability, reduces bugs, and facilitates easier maintenance. This guide explores best practices for writing efficient code, from algorithm optimization to clean coding techniques.

Algorithm Optimization

1. Choosing the Right Algorithm:

- Selecting the most appropriate algorithm for a task is crucial for efficiency. Understanding time and space complexity (Big O notation) helps in evaluating and comparing different algorithms.
- Example: Using a quicksort algorithm for large datasets instead of bubble sort can significantly reduce execution time.

It is works good.

3. Testcase 3: Creating blog for the website **Route:/api/blog/** it is an Post method

Results

Create New Blog

Blog Title:

Alone: Exploring Solitude and Self-Discovery

Blog Content:

A life of an Loner

Blog Author:

AeroKing

Blog Image(Url):

https://img.freepik.com/free-photo/woman-showing-sadness-lonely-place_23-2151597031.jpg?from=1723526386-exp=1723529966-hmac=678dad42cc25a464956315fab7563cf6874b1aa5f53628626c

Blog Body:

Normal B I U

Introduction

Being alone is a universal experience, one that can evoke a wide range of emotions—from peace and contentment to loneliness and despair. In a world that often values constant connection and social interaction, the experience of being alone can be both a challenge and an opportunity for growth. This piece explores the different facets of being alone, the distinction between solitude and loneliness, and the potential for self-discovery in moments of isolation.

Solitude vs. Loneliness

1. Solitude: A Space for Reflection

- Solitude is the state of being alone without feeling lonely. It's a deliberate choice to spend time with oneself, away from the distractions and demands of the outside world. Solitude offers a space for introspection, creativity, and mental clarity.

1. Loneliness: The Pain of Disconnection

- Loneliness, on the other hand, is the feeling of being isolated or disconnected from others. It's the emotional response to a perceived lack of companionship or support, often leading to feelings of sadness, anxiety, and even despair.
- Example: Moving to a new city where you don't know anyone can trigger feelings of loneliness, even in the midst of a bustling environment.

The Benefits of Being Alone

1. Self-Discovery and Personal Growth

- Time spent alone can lead to greater self-awareness and personal growth. Without external influences, individuals can explore their own thoughts, desires, and values, leading to a deeper understanding of themselves.
- Example: A solo trip can be a journey of self-discovery, providing the space to reflect on life goals, passions, and personal identity.

1. Boosting Creativity and Focus

- Solitude often enhances creativity and concentration. Without the noise and distractions of daily life, the mind is free to wander, leading to new ideas and solutions. This focus can be especially beneficial for problem-solving and innovation.
- Example: Many great inventions and artistic works were conceived in moments of solitude, where the creator could fully immerse themselves in their craft.

1. Emotional Resilience and Independence

- Learning to be comfortable alone builds emotional resilience and independence. It teaches self-reliance and the ability to find contentment within oneself, reducing the need for constant external validation.

RetroBlog

Home Create Blog About Us

Blog added successfully

Create New Blog

Blog Title:

Blog Content:

Blog Author:

AeroKing

Blog Image(Uri):

Blog Body:

When blog is added it toast that blog is added.

4. Test case 4: Updating the blog **Route:/api/blog/:id**. it is an patch method

Result

AeroKing

Blog Image(Uri):

Blog Body:

Normal B I U Link List Table Link

=

Update Blog

Blog Updated successfully

Home Create Blog About Us

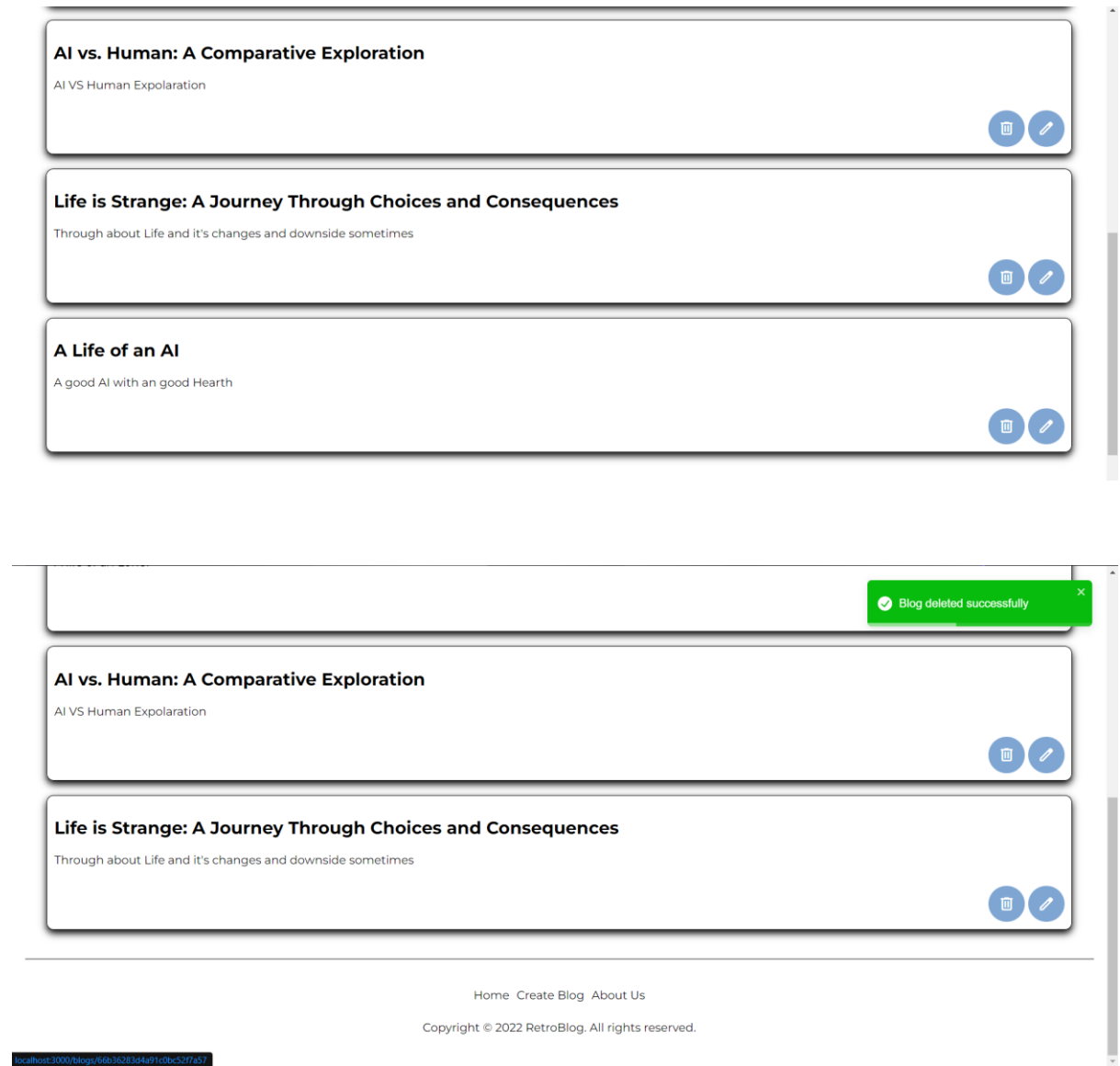
Copyright © 2022 RetroBlog. All rights reserved.

When you update the blog it provides toast blog updated successfully.

5. Testcase 5: Deleting The Blog **Route:/api/bog/:id**. It is an Delete method

Note: You can Access the Delete and Update Route only if you are logged or Sign up in the website and also only for user's written blog . if doesn't create any blogs and he doesn't have access to the update and delete method.

You can only see in the profile



When Delete the blog it toast blog Deleted Successfully.

The unit testcase for Login and Signup:

1. Testcase 1: Login user **Route:/api/post/login**

Result

RetroBlog

Home Create Blog About Us

Login Signup

Login

Enter Your Email

Enter Your Password

Login

If you don't have an account? [SignUp](#)

RetroBlog

Home Create Blog About Us

Logout



Welcome! Aero King

Your email is: Aero@gmail.com

Your Blogs

Alone: Exploring Solitude and Self-Discovery

A life of an Loner



When your logged it has some profile icon on the navbar and the logout button

2. Testcase 2:Sign up user **Route:/api/post/signup**

Result:

RetroBlog

Home Create Blog About Us

Login Signup

Signup

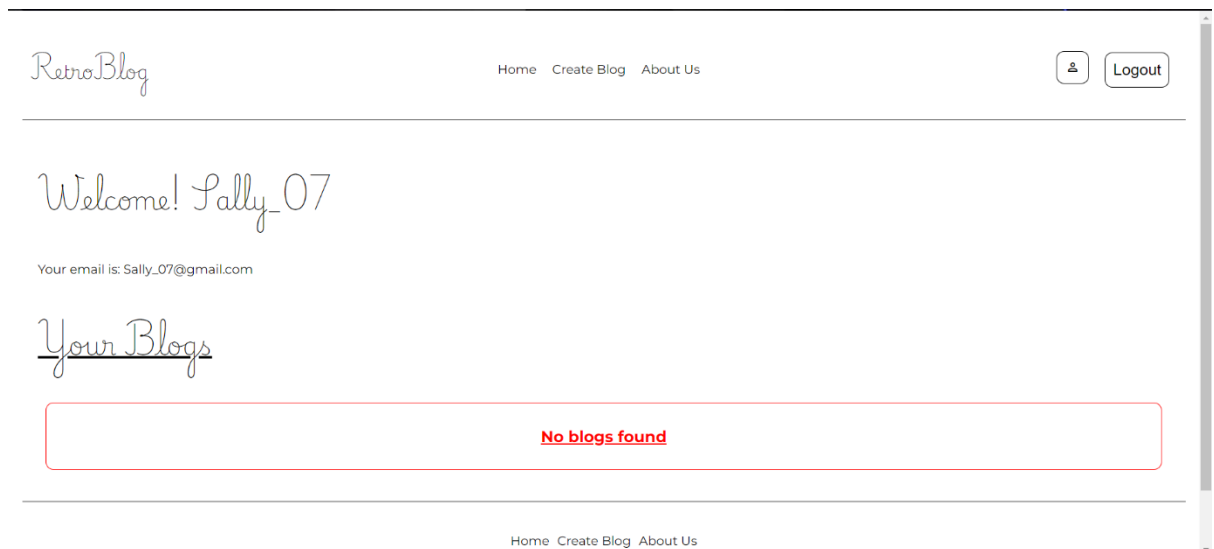
Enter Your Username

Enter Your Email

Enter Your Password

SignUp

If you already have an account? [Login](#)



It same us in the login page.

The unit tests for the Comment components

1. Testcase 1:Fetch all comments for the blog
Route:/api/comment/:Blog_id

Result



2. Testcase 2: fetch one comment from the blog
Route:/api/comment/post/:id

← WRITTEN BY AEROKING

hi man how are you let you know

3. Testcase 3: Post the Blog **Route:/api/comment/**

Result

The image shows two screenshots of the RetroBlog interface. The top screenshot shows the 'Comments' section with a text input field containing 'Hi your good at posting blog' and a blue 'Post' button. The bottom screenshot shows the same section after the comment is posted. A green notification banner at the top right says 'Comment created successfully'. Below it, the comment 'Hi your good at posting blog' is displayed, attributed to 'AeroKing--less than a minute ago'. The comment box is now empty with the placeholder 'Enter your comment'.

Comments

Enter Your Comment

Hi your good at posting blog

Post

Conclusion

Being alone is a multifaceted experience that can lead to both profound self-discovery and, at times, feelings of loneliness. By embracing solitude for personal growth, individuals can transform alone time into a powerful tool for creativity, reflection, and emotional resilience. Balancing solitude with meaningful social connections allows for a richer, more fulfilling life, where one can thrive both alone and with others.

Comment created successfully

Comments

Enter Your Comment

Enter your comment

Post

AeroKing--less than a minute ago

Hi your good at posting blog

When you post an comment it gives comment posted successfully

4. Update the comment **Route:/api/comment/:id**

allows for a richer, more fulfilling life, where one can thrive both alone and with others.

Comments

Enter Your Comment

Hi your good at posting blog

edit

AeroKing--5 minutes ago
Hi your good at posting blog

Home Create Blog About Us

Comments

Enter Your Comment

Enter your comment

Post

AeroKing--11 minutes ago
Hi your good at posting blog thank you

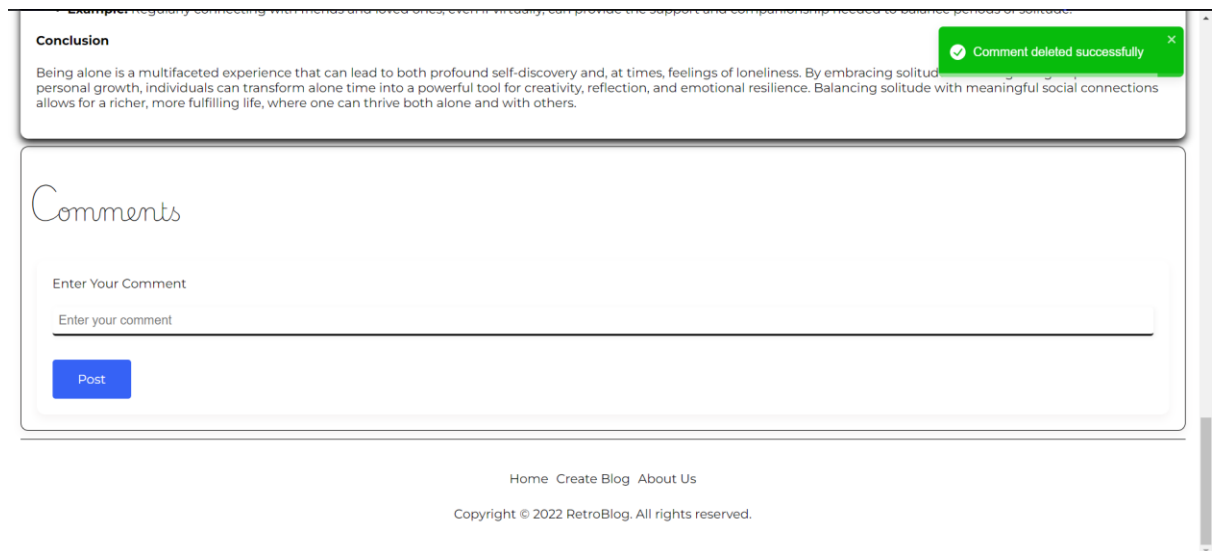
Home Create Blog About Us

Copyright © 2022 RetroBlog. All rights reserved.

Comment Updated successfully

When you update the comment it toast

5. Delete comment **Route:/api/comment/:id**



When you Delete it toast

Note – Delete and Update can be accessed if the user is logged and the user who commented the comment not for the other user's.

Integration test MongoDB integration.

I'm using mongoose to connect the mongodb

First all

I created **.env** file

Which is has three variable which are:

- PORT
- MONGO_URI
- SECRETE

Integration test : connecting mongodb

```
PS F:\Retro Blog\Backend> nodemon dev
[nodemon] 3.1.4
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node dev app.js`
Server started on port 4000
```

I seted if Only the mongodb connection is established then the server is started.