

- Base URL
- Authentication
- API Resources
 - Subscriber
 - Subscribe to Creator
 - Unsubscribe
 - Check Subscription Status
 - Blog
 - Create Blog
 - Manage Blog Media
 - Interactions
 - Comment
 - Add Comment
 - Nested Replies
 - Comment Interactions
 - Category (Admin Only)
 - Create Category
 - List Categories
 - User Interactions
 - Follow System
 - Bookmarks
- Error Handling
- =====bold, italic,underline, headers, links, list, paragraph, text highlight, strikethrough,superscript,subscript, footnotes, Defination List,Task list
- ++++++
- |image, Video, blockquotes,code blocks, tables, blank space|

Blog API Documentation

Base URL

All endpoints are prefixed with `/api`.

Authentication

Endpoints requiring authentication use NextAuth.js. Include the session token in cookies or use the `next-auth.session-token` header.

API Resources

Subscriber

Manage creator subscriptions.

Subscribe to Creator

- **POST** `/subscriber/create/{creatorId}`
 - **Body:** `{ email?: string }` (Required if unauthenticated)
 - **Success:** 200 OK with subscription confirmation
 - **Error:** 400 if missing parameters, 500 on server error

Unsubscribe

- **DELETE** `/subscriber/delete/{subscriberId}`
 - **Headers:** Valid session token
 - **Success:** 200 OK with success message
 - **Error:** 401 if unauthorized, 404 if subscriber not found

Check Subscription Status

- **GET** `/subscriber/read/{creatorId}/{email}`
 - **Response:** `{ isSubscribed: boolean }`
 - **Error:** 400 if missing IDs

Blog

CRUD operations for blog posts and interactions.

Create Blog

- **POST** /blog/create

- **Body:**

```
{
  "title": "Post Title",
  "contentBody": "Content...",
  "isPublished": true,
  "category": "categoryId",
  "tags": ["tag1", "tag2"],
  "publishDateTime": "2024-01-01T00:00:00Z"
}
```

- **Success:** 201 Created with blog data

Manage Blog Media

- **Upload:** POST /blog/media/upload/{blogId}
- **Read:** GET /blog/media/read/{blogId}/{mediaId}
- **Update:** PUT /blog/media/update/{blogId}/{mediaId}
- **Delete:** DELETE /blog/media/remove/{blogId}/{mediaId}

Interactions

- **Like:** PUT /blog/like/{blogId}
 - **Response:** { likeCount: number, likeStatus: boolean }
- **Share:** PUT /blog/share/{blogId}
 - **Response:** Updated share count

Comment

Manage blog post comments.

Add Comment

- **POST** /comment/create/{blogId}
 - **Body:** { comment: "Comment text" }
 - **Success:** 200 with comment data

Nested Replies

Comments support nested replies through recursive population.

Comment Interactions

- **Like:** PUT `/comment/likes/{blogId}/{commentId}`
 - **Response:** Updated like status and count
-

Category (Admin Only)

Category management (requires admin privileges).

Create Category

- **POST** `/category/create`
 - **Body:** `{ name: "Category Name" }`

List Categories

- **GET** `/category/read`
 - **Response:** Array of category objects
-

User Interactions

Follow System

- **Follow/Unfollow:** PUT `/user/follow/{targetUserId}`
 - **Response:**

```
{
  "followStatus": boolean,
  "followingCount": number,
  "followerCount": number
}
```

Bookmarks

- **Toggle Bookmark:** PUT `/user/bookmark/{contentId}`

- **Response:**

```
{
  "bookmarkStatus": boolean,
  "bookmarkCount": number
}
```

Error Handling

Standard error response format:

```
{
  "success": false,
  "message": "Error message",
  "error": { ...error details }
}
```

=====

===== **bold**,
italic,underline, headers, links, list,
paragraph, text highlight, strikethrough,
superscript,subscript, footnotes,
Defination List, Task list

+++++

+++++

|image, Video, blockquotes,code blocks,
tables, blank space|

```
print("rahul")  
c=int(input("Enter a number"))
```
