

Blog APIs

All the APIs for the blog app.

Contact: Kunal Bhavsar (bhavsarkunal2103@gmail.com)

Authentication

All API endpoints require authentication using JWT. Include the JWT token in the **Authorization** header.

Servers

- Development: <http://localhost:8080>
- Testing: <http://localhost:8080>

Security Scheme

- **Auth:** JWT token in the **Authorization** header.

APIs

SWAGGER APIS

```
"url" "http://localhost:8080/v3/api-docs"
"description" "Swagger API Documentation"

"url" "http://localhost:8080/swagger-ui/index.html"
"description" "Swagger UI"
```

Users

Update User

- **Endpoint:** `/api/users/update/{id}`
- **Method:** `PUT`
- **Description:** Update user details by ID.
- **Request Body:** `UserDto`
- **Responses:**
 - `200 OK`: `UserDto`

Create User

- **Endpoint:** `/api/users/create`
- **Method:** `POST`
- **Description:** Create a new user.
- **Request Body:** `UserDto`
- **Responses:**
 - `200 OK`: `UserDto`

Get All Users

- **Endpoint:** `/api/users/get`
- **Method:** `GET`
- **Description:** Get all users.
- **Responses:**
 - `200 OK`: Array of `UserDto`

Get User By ID

- **Endpoint:** `/api/users/get/{id}`
- **Method:** `GET`
- **Description:** Get user by ID.
- **Responses:**
 - `200 OK`: `UserDto`

Delete User

- **Endpoint:** `/api/users/delete/{id}`
- **Method:** `DELETE`
- **Description:** Delete user by ID.
- **Responses:**
 - `200 OK`

Posts

Update Post

- **Endpoint:** `/api/posts/update/{id}`
- **Method:** `PUT`
- **Description:** Update post details by ID.
- **Request Body:** `PostDto`
- **Responses:**
 - `200 OK`: `PostDto`

Create Post

- **Endpoint:** `/api/posts/create`
- **Method:** `POST`
- **Description:** Create a new post.
- **Request Body:** `PostDto`
- **Responses:**
 - `200 OK`: `PostDto`

Get All Posts

- **Endpoint:** `/api/posts/get`
- **Method:** `GET`
- **Description:** Get all posts with pagination support.
- **Parameters:**
 - `pageNo` (optional)
 - `pageSize` (optional)
 - `sortField` (optional)
 - `sortType` (optional)
- **Responses:**
 - `200 OK`: `PostResponse`

Get Post By ID

- **Endpoint:** `/api/posts/get/{id}`
- **Method:** `GET`
- **Description:** Get post by ID.
- **Responses:**
 - `200 OK`: `PostDto`

Get Post By User ID

- **Endpoint:** `/api/posts/getByUser`
- **Method:** `GET`
- **Description:** Get posts by user ID.
- **Parameters:**
 - `userId`
- **Responses:**
 - `200 OK`: Array of `PostDto`

Get Post By Category ID

- **Endpoint:** `/api/posts/getByCategory`
- **Method:** `GET`
- **Description:** Get posts by category ID.
- **Parameters:**
 - `categoryId`
- **Responses:**
 - `200 OK`: Array of PostDto

Search Post

- **Endpoint:** `/api/posts/search/{keywords}`
- **Method:** `GET`
- **Description:** Search posts by keywords.
- **Parameters:**
 - `keywords`
- **Responses:**
 - `200 OK`: Array of PostDto

Delete Post

- **Endpoint:** `/api/posts/delete/{id}`
- **Method:** `DELETE`
- **Description:** Delete post by ID.
- **Responses:**
 - `200 OK`

Comments

Update Comment

- **Endpoint:** `/api/comments/update/{id}`
- **Method:** `PUT`
- **Description:** Update comment by ID.
- **Request Body:** CommentDto
- **Responses:**
 - `200 OK`: CommentDto

Create Comment

- **Endpoint:** `/api/comments/upload`
- **Method:** `POST`
- **Description:** Create a new comment.

- **Request Body:** CommentDto
- **Responses:**
 - 200 OK: CommentDto

Get Comments

- **Endpoint:** /api/comments/get
- **Method:** GET
- **Description:** Get comments with optional filtering by user ID or post ID.
- **Parameters:**
 - userId (optional)
 - postId (optional)
- **Responses:**
 - 200 OK: Array of CommentDto

Get Comment By ID

- **Endpoint:** /api/comments/get/{id}
- **Method:** GET
- **Description:** Get comment by ID.
- **Responses:**
 - 200 OK: CommentDto

Delete Comment

- **Endpoint:** /api/comments/delete/{id}
- **Method:** DELETE
- **Description:** Delete comment by ID.
- **Responses:**
 - 200 OK

Categories

Update Category

- **Endpoint:** /api/categories/update/{id}
- **Method:** PUT
- **Description:** Update category by ID.
- **Request Body:** CategoryDto
- **Responses:**
 - 200 OK: CategoryDto

Create Category

- **Endpoint:** `/api/categories/create`
- **Method:** `POST`
- **Description:** Create a new category.
- **Request Body:** CategoryDto
- **Responses:**
 - `200 OK`: CategoryDto

Get All Categories

- **Endpoint:** `/api/categories/get`
- **Method:** `GET`
- **Description:** Get all categories.
- **Responses:**
 - `200 OK`: Array of CategoryDto

Get Category By ID

- **Endpoint:** `/api/categories/get/{id}`
- **Method:** `GET`
- **Description:** Get category by ID.
- **Responses:**
 - `200 OK`: CategoryDto

Delete Category

- **Endpoint:** `/api/categories/delete/{id}`
- **Method:** `DELETE`
- **Description:** Delete category by ID.
- **Responses:**
 - `200 OK`

File Service

Upload File

- **Endpoint:** `/api/file/upload`
- **Method:** `POST`
- **Description:** Upload a file with optional association to a post.
- **Parameters:**
 - `postId` (optional)
- **Request Body:** Binary file
- **Responses:**
 - `200 OK`: PostDto (updated with file information)

Download File

- **Endpoint:** `/api/file/download`
- **Method:** `GET`
- **Description:** Download a file by filename.
- **Parameters:**
 - `fileName`
- **Responses:**
 - `200 OK`