

Blog API Documentation

Overview

This API allows users to register, login, create, read, update, and delete blog posts and comments. The application uses JWT for authentication, SQLAlchemy for ORM, and SQLite as the database.

Installation

```
pip install Flask Flask-SQLAlchemy Flask-JWT-Extended
```

Configuration

- `SQLALCHEMY_DATABASE_URI`: Database URI for SQLAlchemy.
- `SQLALCHEMY_TRACK_MODIFICATIONS`: Set to False to disable tracking modifications.
- `JWT_SECRET_KEY`: Secret key for JWT.

Models

User

- `id`: Integer, primary key.
- `username`: String(80), unique, not nullable.
- `email`: String(120), unique, not nullable.
- `password`: String(200), not nullable.

Post

- `id`: Integer, primary key.
- `title`: String(100), not nullable.
- `content`: Text, not nullable.
- `author_id`: Integer, foreign key to user.id, not nullable.
- `created_at`: DateTime, default is the current time.

Comment

- id: Integer, primary key.
- post_id: Integer, foreign key to post.id, not nullable.
- content: Text, not nullable.
- author_id: Integer, foreign key to user.id, not nullable.
- created_at: DateTime, default is the current time.
- post: Relationship to Post.
- author: Relationship to User.

Endpoints

Register User

- URL: /register
- Method: POST
- Authentication: None
- Description: Registers a new user.

Create Post

- URL: /posts
- Method: POST
- Authentication: JWT required
- Description: Creates a new post.

Get All Posts

- URL: /posts
- Method: GET
- Authentication: None
- Description: Retrieves all posts.

Get Single Post

- URL: /posts/<int:post_id>
- Method: GET
- Authentication: None
- Description: Retrieves a single post by its ID.

Update Post

- URL: /posts/<int:post_id>
- Method: PUT
- Authentication: JWT required
- Description: Updates an existing post. Only the author can update the post.

Delete Post

- URL: /posts/<int:post_id>
- Method: DELETE
- Authentication: JWT required
- Description: Deletes an existing post. Only the author can delete the post.

Create Comment

- URL: /comments
- Method: POST
- Authentication: JWT required
- Description: Creates a new comment.

Get Comments for a Post

- URL: /comments
- Method: GET
- Authentication: None
- Description: Retrieves all comments for a specific post.
- Query Parameters:
 - post_id: ID of the post to retrieve comments for.

Update Comment

- URL: /comments/<int:comment_id>
- Method: PUT
- Authentication: JWT required
- Description: Updates an existing comment. Only the author can update the comment.

Delete Comment

- URL: /comments/<int:comment_id>
- Method: DELETE
- Authentication: JWT required
- Description: Deletes an existing comment. Only the author can delete the comment.

Running the Application

`python <filename>.py`

Error Handling

- 404 Not Found: If a resource is not found (e.g., a post or comment), the API returns a 404 status code.
- 403 Forbidden: If a user attempts to modify or delete a resource they do not own, the API returns a 403 status code.
- 401 Unauthorized: If a user attempts to access a resource that requires authentication without a valid token, the API returns a 401 status code.