APIs

Blog

Category

- **Create a Category**
    - URL: `/api/categories`
    - Method: `POST`
    - Payload:

      ```
      json
      Copy code
      {
        "name": "Technology",
        "slug": "technology"
      }
      ```

- **Get All Categories**
    - URL: `/api/categories`
    - Method: `GET`
- **Update a Category**
    - URL: `/api/categories/<id>`
    - Method: `PUT`
    - Payload:

      ```
      json
      Copy code
      {
        "name": "Tech",
        "slug": "tech"
      }
      ```

- **Delete a Category**
    - URL: `/api/categories/<id>`
    - Method: `DELETE`

Tag

- **Create a Tag**
    - URL: `/api/tags`
    - Method: `POST`
    - Payload:

      ```
      json
      Copy code
      {
        "name": "Flask",
      ```

```
  "slug": "flask"
}
```

- **Get All Tags**
  - URL: `/api/tags`
  - Method: `GET`
- **Update a Tag**
  - URL: `/api/tags/<id>`
  - Method: `PUT`
  - Payload:

    ```
    json
    Copy code
    {
      "name": "Flask Framework",
      "slug": "flask-framework"
    }
    ```

- **Delete a Tag**
  - URL: `/api/tags/<id>`
  - Method: `DELETE`
- **Create a Blog Post**
  - URL: `/api/blogs`
  - Method: `POST`
  - Payload:

    ```
    json
    Copy code
    {
      "title": "Sample Blog Post",
      "slug": "sample-blog-post",
      "content": "This is the content of the sample blog post.",
      "category_id": 1,
      "user_id": 1,
      "tags": [1, 2]
    }
    ```

- **Get a Blog Post**
  - URL: `/api/blogs/<id>`
  - Method: `GET`
- **Update a Blog Post**
  - URL: `/api/blogs/<id>`
  - Method: `PUT`
  - Payload:

    ```
    json
    Copy code
    {
      "title": "Updated Blog Post",
      "content": "Updated content of the blog post."
    ```

```
}
```

- **Delete a Blog Post**
  - URL: `/api/blogs/<id>`
  - Method: `DELETE`

Folder Structure

- `app/`: Contains the main application code.
- `migrations/`: Flask-Migrate migrations.
- config..py: Flask and PostgreSQL Database configuration.
- `docker-compose.yml`: Docker Compose configuration.
- `requirements.txt`: Python dependencies.
- run.py: To run the Project

Database Models

- **User**: Represents a registered user with username and email.
- **Category**: Represents a category for blog posts.
- **Tag**: Represents tags associated with blog posts.
- **Post**: Represents a blog post with title, content, and related data.
- **Comment**: Represents comments on blog posts.