

1. Problem Statement:

Creating a simple Node.js backend application for managing a list of users and their posts involves users creating posts, with each post being associated with a specific user.

2. Requirements/Features:

- Create a MySQL database with two tables: `users` and `posts`.
- Define a foreign key relationship between the `users` and `posts` tables.
- Implement CRUD operations for users and posts.
- Validate and sanitize user inputs.
- Implement pagination for fetching posts.
- Handle errors gracefully with proper error handling.

3. Technologies:

- Node.js
- Express.js
- MySQL
- Sequelize (for ORM)
- Express-validator for input validation
- Pagination for fetching posts

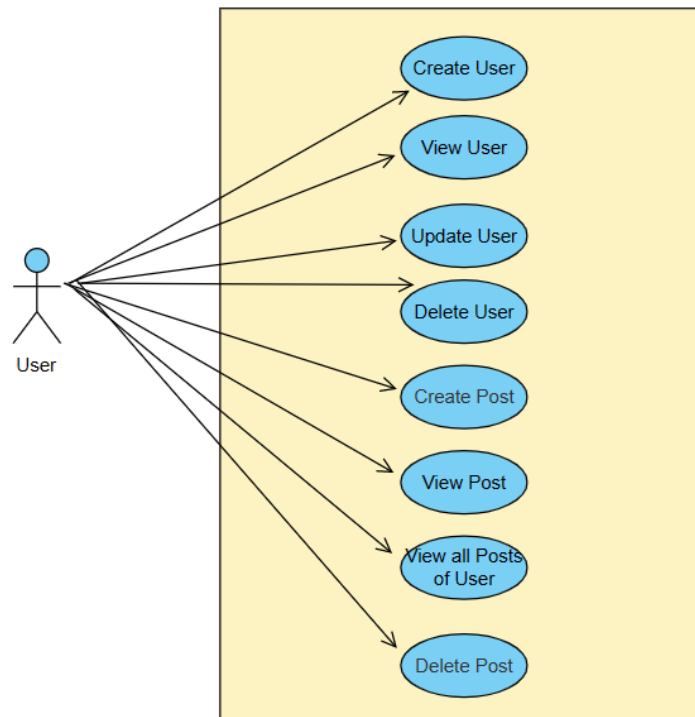
4. APIs to be Developed:

- User API:
 - Create a user
 - Get a user by ID
 - Update a user
 - Delete a user
- Post API:
 - Create a post
 - Get a post by ID
 - Get all posts by a user
 - Update a post
 - Delete a post

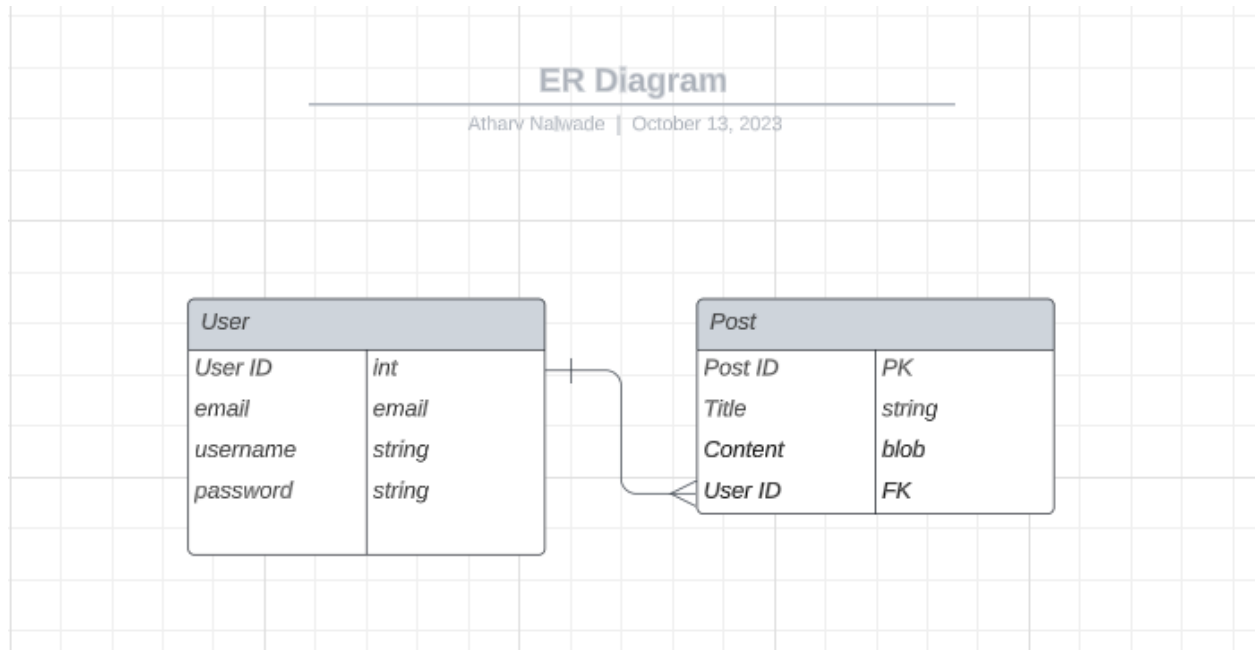
5. Modules:

- User module
- Post module
- Database module
- Error handling module
- Validation module

6. Use Case Diagram



7. Class Diagram:



8. Future Scope:

- Implement user roles and permissions if required in the future.
- Add user profile images.
- Implement search functionality for posts.
- Improve security by implementing rate limiting and request validation.
- Implement unit testing and integration testing.
- Deploy the application to a cloud platform (e.g., AWS, Heroku).
- Implement real-time features using WebSocket (e.g., chat functionality).