



## **Task Title: RESTful API with Authentication**

### **Task Description:**

You are tasked with building a RESTful API using Node.js and Express.js or Nest.js for a simple online task management system. The API should have the following functionality:

1. User registration: Users can create an account by providing a username, email, and password.
2. User login: Registered users can log in with their email and password to access their tasks.
3. Task CRUD operations: Once logged in, users can perform the following operations on their tasks:
  - Create a new task with a title and description.
  - Retrieve a list of their tasks.
  - Retrieve a single task by its unique ID.
  - Update an existing task by its ID.
  - Delete a task by its ID.
4. User authentication: Implement authentication middleware to ensure that only authenticated users can access task-related endpoints.
5. Use JWT (JSON Web Tokens) for authentication and authorization.
6. Secure password storage: Hash and salt user passwords before storing them in the database.

### **Requirements:**

- Use Nest.js or Express.js for routing and middleware.
- Implement data storage using a database of your choice (e.g., MongoDB, MySQL).
- Use appropriate HTTP methods (POST, GET, PUT, DELETE) and status codes.
- Include validation for input data (e.g., validate email format, required fields for tasks).
- Write clear and well-organized code with comments where necessary.
- Create a README.md file explaining how to set up and run the API, including how to use authentication.

### **Bonus Points:**

- Implement pagination and sorting options for listing tasks.
- Add user-friendly error handling and return meaningful error messages.
- Write unit tests for API endpoints using a testing framework like Jest.
- Implement role-based access control (e.g., admin vs. regular user).
- Provide API documentation using tools like Swagger or Postman.
- Dockerize the application for easy deployment.

This task will assess the candidate's ability to create a secure and well-structured API, handle user authentication, and work with databases and JWT for authorization. Good luck with your evaluation!