

Task Requirements:

1. Set up a .NET Core API project with EF Code First, JWT Authentication & Authorization.
2. Consume the Todos API from "<https://jsonplaceholder.typicode.com/todos>" to retrieve the Todos data. (You can use [Pagination - Bonus](#))
3. Implement CRUD operations for the Todos entity:
 - Create a new Todo item.
 - Retrieve a specific Todo item by its ID.
 - Update an existing Todo item.
 - Delete a Todo item.
4. [Implement the Repository and Unit of Work pattern:](#)
 - Create a separate repository class that encapsulates data access operations for the Todos entity.
 - Implement methods in the repository for CRUD operations and querying Todo items.
 - Create a Unit of Work class that manages transactions and coordinates multiple repository operations within a single transaction scope.
5. [Use a mapping library \(e.g., AutoMapper\) to map between domain models and DTOs:](#)
 - Define mapping profiles to configure the mappings between domain models (e.g., Todo) and DTOs used in the API.
6. Secure the CRUD endpoints using JWT authentication.
7. Test the API endpoints using tools like Swagger or Postman.

Steps 4, 5 and Pagination are additional (BONUS).

Expect to see :

- **Account Controller → Login & Register Endpoints**
- **ToDo Controller → Create,Read,Update,Delete Endpoint**
- **LiveToDo Controller → Read Endpoint (Data directly from <https://jsonplaceholder.typicode.com/todos>)**

Deadline: 20-8-2023 4PM