Brief Documentation

Each Layer in the Project

1. Core Layer (TaskManagement.Core)
This layer contains the main entity and interface definitions. It defines the **TaskItem** entity with its properties and the **ITaskRepository** interface that declares CRUD operations. It focuses only on the structure of the application and does not include any implementation details.

2. Infrastructure Layer (TaskManagement.Infrastructure)
This layer manages all data persistence and database-related logic. It includes the **TaskDbContext** for EF Core configuration and **TaskRepository** which implements **ITaskRepository**. The repository handles Create, Read, Update, and Delete operations using EF Core and communicates directly with the database.

How the Repository is Implemented
The repository follows the Repository Pattern. The interface defines the operations, and the concrete class (**TaskRepository**) provides the EF Core-based implementation. This separation keeps the system clean, testable, and maintainable.

How CRUD Operations Work
Each operation interacts with the repository:
- **Create:** Adds new task data to the database.
- **Read:** Retrieves tasks or specific records.
- **Update:** Modifies existing task details.
- **Delete:** Removes records from the database.
This flow ensures clear separation between UI and data access.

Database Configuration
The database is configured with SQL Server and Entity Framework Core. Migrations are used to create and manage tables automatically. The **TaskDbContext** class ensures proper mapping and configuration of entities.

Summary
The Task Management System is structured using a clean N-Tier architecture with distinct layers for core logic and data persistence. It efficiently applies the Repository Pattern with EF Core to ensure modularity, scalability, and easy maintenance.