

# Project Overview

**Title:** Product Manage API

**Description:** A RESTful API for managing product inventory, including CRUD operations, stock management, and unique product ID generation in a distributed environment.

**Repo:** <https://github.com/yash535/ProductManager>

**Clone:** `git clone https://github.com/yash535/ProductManager`

## **Key Features:**

- Create, read, update, and delete products
- Manage product stock levels
- Support for auto-incremented product IDs
- Handling of stock adjustments (both decrement and increment)
- Error handling for stock levels and product existence
- Used EF Code First Migration

## **Technology Stack:**

- ASP.NET Core
- .NET SDK 8.0
- Entity Framework Core
- Azure SQL Database
- In-Memory Database (for testing)
- NUnit (for unit testing)
- Moq (for mocking dependencies in tests)
- Visual Studio 2022 or later (or any other preferred IDE)

# Architecture:

## High-Level Design:

**API Layer:** Handles HTTP requests and responses. Contains controllers that interact with the service layer.

**Service Layer:** Manages business logic and interacts with the repository layer.

**Repository Layer:** Handles data access and operations with the database context.

**Data Layer:** Uses Entity Framework Core to interact with the database.

## Design Patterns:

**Repository Pattern:** Encapsulates data access logic in the 'ProductRepository' class.

**Unit of Work Pattern:** Managed through the 'ProductContext' class

**Dependency Injection:** Used to inject repository instances into controllers.

## Model:

```
public class Product
{
    public int Id { get; set; }
    public string Name { get; set; }
    public decimal Price { get; set; }
    public int StockAvailable { get; set; }
    public string Description { get; set; }
}
```