appdemo

Aplicación demo para aplicación de metodología modular

La aplicación puedes ser ejecutada de varias maneras para propósitos de desarrollo y/o prueba.

Emplear una Base de Datos en memoria

Escenario facilita la ejecución del aplicativo en modo de prueba O desarrollo de forma que no requerirá hacer ningún ajuste pues toda la configuración del aplicativo está realizada por defecto dentro del repositorio.

Ejecución

- Ubicado en appdemo\BackEnd\App.Api
- 2. Ejecute dotnet build y luego Ejecute dotnet run
- 3. Verifique que puede obtener un resultado en alguna de las siguientes rutas
- https://localhost:5001/api/users
- http://localhost:5000/api/users

Empleando Docker

Para facilitar el despliegue y prueba de la aplicación se puede apoyar del archivo docker-compose ubicado en la raíz de repositorio en el se crea un servicio de base de datos para el cual ya se han definido nombres de base de datos y contraseñas compatibles con el aplicativo de no de manera que no se requiere hacer ningún ajuste

Ajustes previos

Ajuste el archivo BackEnd\App.Api\ServiceExtensions.cs para utilice la sentencia:

 $options Builder. Use Sql Server (connection String); \ // \ Sql Server$

en vez de:

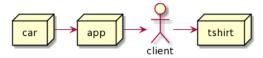
optionsBuilder.UseInMemoryDatabase("db_memoria"); // En memoria

o viceversa, según sea el caso de su preferencia.

Ejecución

- 1. Ubicado en appdemo
- 2. Ejecute docker-compose up db
- Ubiquese en appdemo\BackEnd\App.Api
- 4. Ejecute dotnet build y luego Ejecute dotnet run
- 5. Verifique que puede obtener un resultado en alguna de las siguientes rutas
- https://localhost:5001/api/users
- http://localhost:5000/api/users

Arquitectura



Visión General (appdemo)

Welcome to PlantUML!

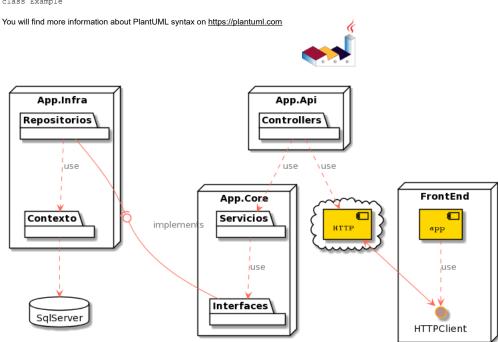
If you use this software, you accept its license. (details by typing license keyword)

You can start with a simple UML Diagram like:

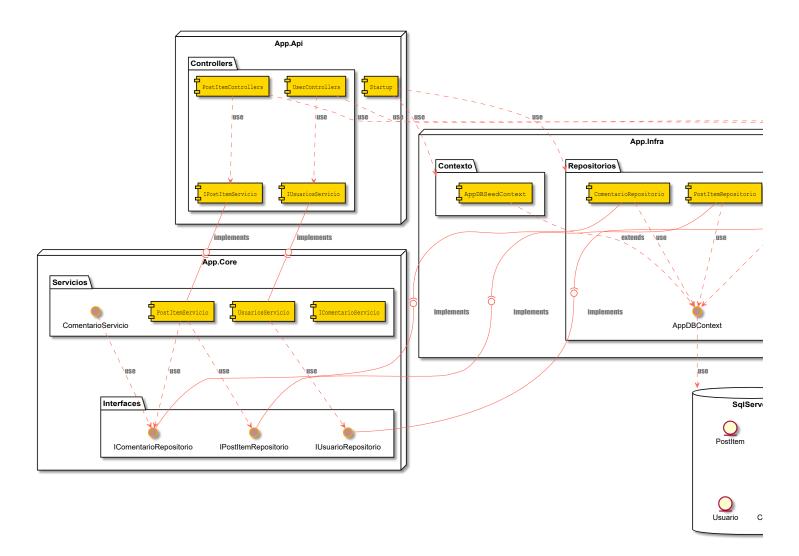
Bob->Alice: Hello

Or

class Example



Vista Detallada (appdemo)



Referencias

- UML conventions
- Component UML Diagrams Plantuml
- Proxy render UML
- plantuml-styles renders