Coding Challenge: Calculator Service

Johan Alexis Ruiz Echavarria

1. Introducción

Este documento tiene como objetivo dar detalle de los diferentes aspectos técnicos de la solución desarrollada para el reto.

A continuación, se da explicación del diseño arquitectónico de la solución, despliegue y funcionamiento.

1. Arquitectura
   1. Tecnologías utilizadas:

* .NET Framework 4.6
* ASP.NET
* Entity Framework 6.2
* SQL Server
* Unity: utilizado para la inyección de dependencias e inversión del control
  1. Prerrequisitos:
* Disponibilidad de servidor de base de datos Microsoft SQL Server
* .NET Framework 4.6
  1. Estructura de la solución (modelo de despliegue):

La solución consiste principalmente en un sistema cliente - servidor donde el cliente está representado por una aplicación de consola que sirve de componente de interacción con el usuario y un servidor consistente en un Web API, el cual es el encargado de ofrecer las capacidades de negocio de la aplicación, en este caso las operaciones aritméticas definidas en el reto y la opción de consulta de operaciones registradas.

* 1. Arquitectura

A pesar de la simplicidad del sistema solicitado y con el objetivo de mostrar las capacidades de desarrollo de una aplicación altamente estructurada y con características de un sistema de información robusto y dirigido por drivers de arquitectura, escalable con bajo acoplamiento y alta cohesión por ejemplo, se determinó adoptar DDD (diseño dirigido por el dominio) como marco de referencia en la definición de la arquitectura de la aplicación.

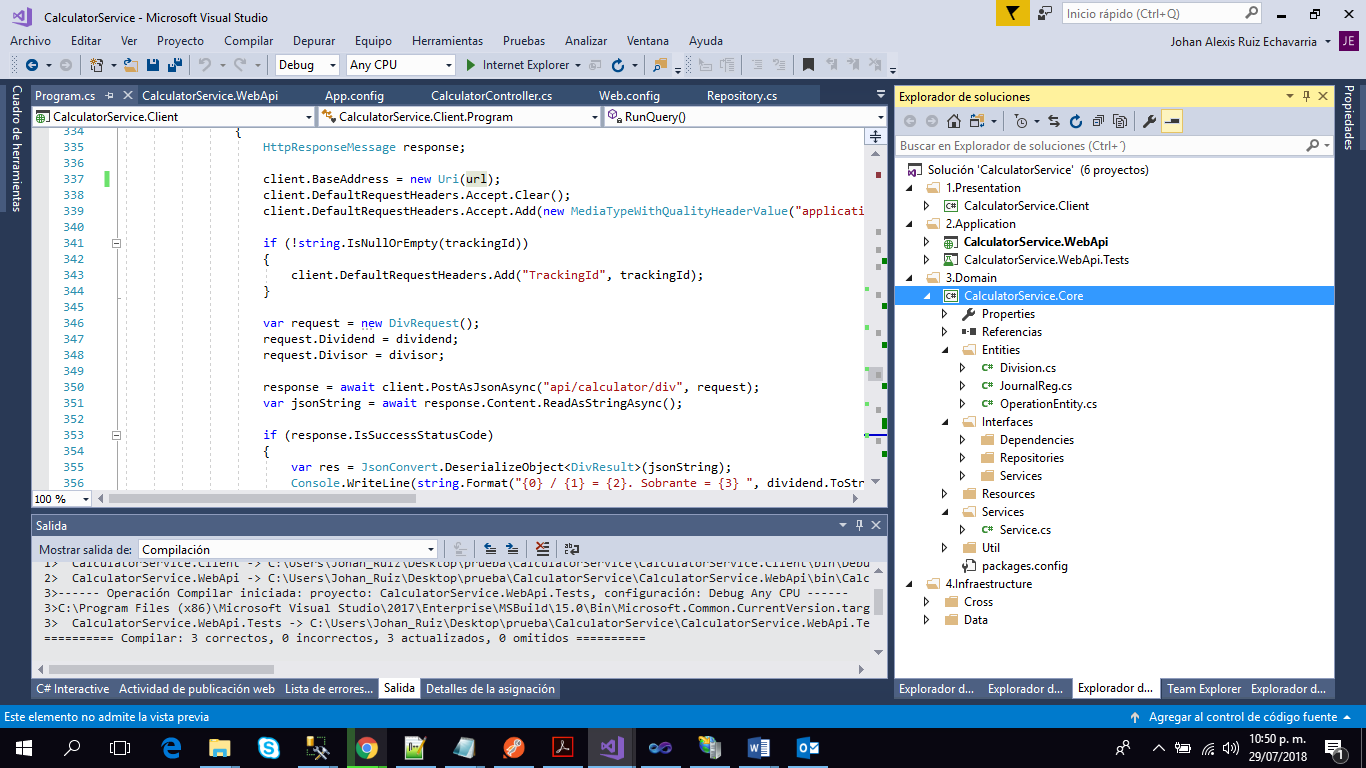
De acuerdo con lo anterior, la solución cuenta con componentes diferenciados en las siguientes capas:

**- Capa de Presentación:** componentes de front - end, en este caso representado por la aplicación de consola.

**- Capa de aplicación:** componentes livianos (sin lógica de negocio) que se encargan de poner a disposición las capacidades de negocio ofrecidas por el sistema (componente de negocio). En este caso el componente consiste en un web API que expone a través de sus endpoints las operaciones matemáticas y de consulta requeridas por el sistema.

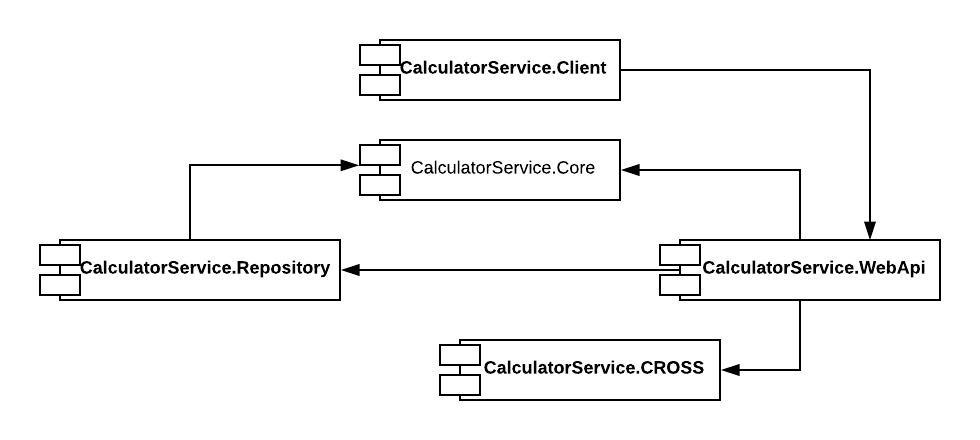
**- Capa de negocio:** Componentes que representan el CORE del negocio, estos contienen las entidades y lógica de negocio de la aplicación, o bien llamados Servicios. En esta solución, este componente es un assembly que contiene toda la lógica de las operaciones matemáticas de la calculadora, las restricciones de acuerdo con cada formula algebraica y validación de datos.

Por otro lado, en este componente también se define las interfaces de los repositorios y servicios de negocio con que cuenta la aplicación, esto permite implementar la inyección de dependencias e inversión de control. La aplicación de este patrón permite mayor modularidad de la aplicación y facilita la implementación de pruebas unitarias.



**- Capa de infraestructura:** Esta capa incluye principalmente los componentes periféricos de la aplicación, tal como los encargados de la persistencia de datos y aquellos transversales a toda la solución como utilitarios de seguridad y generadores de logs técnicos. El acceso a estos se da a través de la implementación de las interfaces definidas en la capa de negocio.

En este caso, contamos con dos componentes en esta capa, uno llamado “Repository” que es el encargado de las operaciones de acceso a la base de datos y otro llamado “CROSS” que se encarga de realizar el registro en un log para registro de excepciones.



1. Persistencia de datos
   1. Base de datos

Como repositorio de base de datos se definió el registro en una tabla en SQL Server llamada “journal”, la cual es creada automáticamente por la aplicación gracias a la implementación de Code First de Entity Framework. En esta estructura se almacena la información de las operaciones realizadas con un identificador de usuario.

* 1. Log técnico

Para el registro de información de seguimiento de la aplicación o log técnico, se utilizó la librería log4Net, la cual facilita la generación archivos planos para registrar la información que sea necesaria, en este caso la descripción de las excepciones del sistema y las validaciones de negocio fallidas.

La aplicación está configurada para que almacene el log de manera diaria en la carpeta C:\Windows\Temp\CalculatorService.