Jeysson Stevens Ramírez Vargas

Jeysson.ramirez@outlook.com

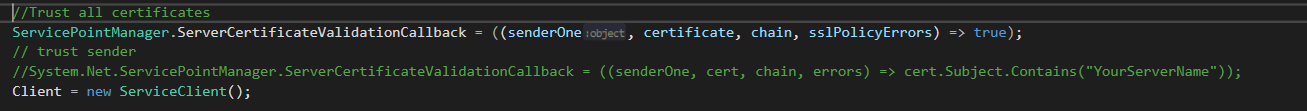
Info

A continuación, se hace entrega de la documentación y explicación del desarrollo hecho para la prueba presentada

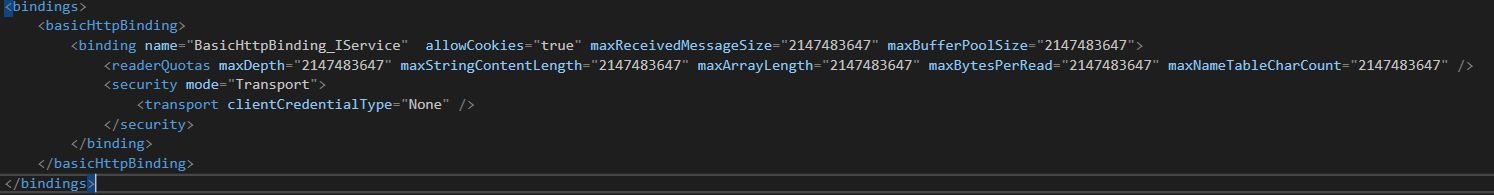
Documentacion Desarrollo

Prueba de Ingreso

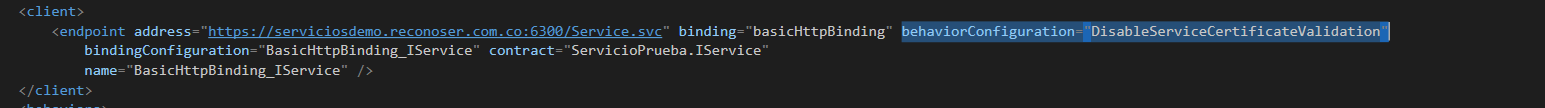
**1: Hacer funcionar la aplicación. Debido al aumento de transacciones y al colocar al servicio con SSL la aplicación actual está fallando.**



Agregar Delegado que omite la validación del certificado SSL



Agregar Configuración de Binding para permitir el Uso del Protocolo HTTPS

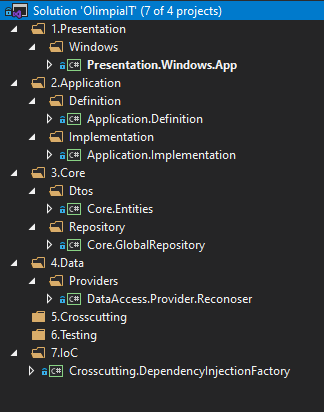


Usar Configuración en el Endpoint

**2: Estructurar mejor el código. Uso de patrones, buenas prácticas, etc. OK**

**Use una arquitectura N Niveles**

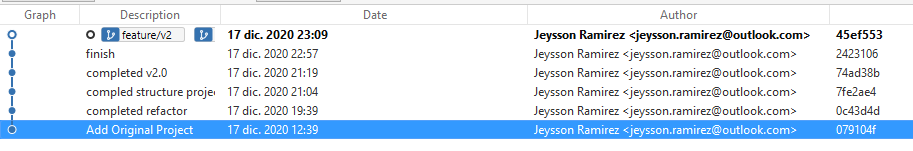
1. Capa de presentación sin Lógica
2. Capa de lógica de Servicios
3. Capa Core de Entidades y clases de Definición del Negocio
4. Capa de Acceso a datos (se movió la referencia del servicio a esta capa)
5. Capa de Transversal uso de Logging y manejo de errores (No se uso)
6. Capa de Proyectos de Testing (No se uso)
7. Capa de Inyección de Dependencias (Unity).



## Patrones Implementados

* Inversión de dependencias
* Fachada
* Dependencia de definiciones y no de implementaciones
* Uso de delegados para actualización de interfaz
* Se actualizo el Framework del 4.6.2 a 4.72
* Patron Repository

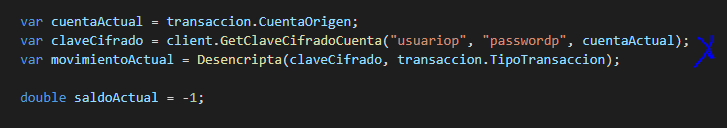
## Uso de GitFlow



**3: Optimizar el código, como se mencionó en el paso 1 el aumento de transacciones ha causado que el cálculo de los saldos se demore demasiado. OK**

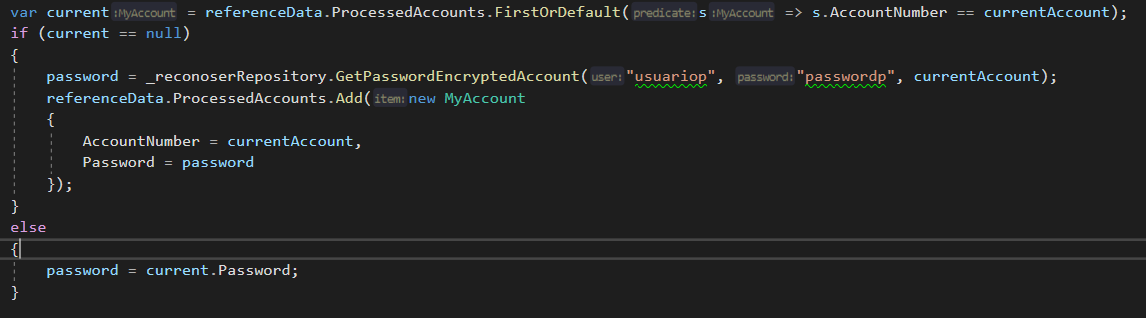
Se detecto que en la línea 48 del archivo Form.cs se está consultando siempre el servicio para extraer la contraseña para descifrar los valores en la iteración

**Rendimiento 132 seg**



Se procede a almacenar la relación de las contraseñas por cuenta y primero consulta si ya se tiene este dato antes de ir al servicio a obtenerlo

**Rendimiento 18 seg**



**Se Crearon dos métodos para el procesamiento de la información**

* Uno de uso simple sobre toda la información (arreglo)
* Uno que hilos del computador dividiendo la información en arreglos y procesados en simultaneo, pero genero problema con la actualización del interfaz (barra de progreso)
* La versión Original del código entregado está en **“OlimpiaItTest\src\WindowsLiderEntrega”**
* La versión nueva del código está en

**“OlimpiaItTest\src\OlimpiaIT”**

**4: Adicionar una barra de progreso al formulario. Actualizar la barra con el progreso del proceso, evitando bloqueos del GUI.**

Se agrego la barra de proceso actualizada por medio de un delegado sin generar bloqueo en la interfaz

