

ACTUALIZACIÓN DEL SOFTWARE FINANCIAL BUSINESS SYSTEM DE LA
CORPORACIÓN NARIÑO EMPRESA Y FUTURO CONTACTAR MEDIANTE SU
MIGRACIÓN A LA TECNOLOGÍA WINDOWS COMMUNICATION FOUNDATION

JONNATHAN ANDRES RAMIREZ IGUA

UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA INGENIERÍA DE SISTEMAS
SAN JUAN DE PASTO
2016

ACTUALIZACION DEL SOFTWARE FINANCIAL BUSINESS SYSTEM DE LA
CORPORACIÓN NARIÑO EMPRESA Y FUTURO CONTACTAR MEDIANTE SU
MIGRACIÓN A LA TECNOLOGÍA WINDOWS COMMUNICATION FOUNDATION

JONNATHAN ANDRES RAMIREZ IGUA

Trabajo de Grado presentado como requisito parcial para optar al título de
Ingeniero de Sistemas

ASESOR:

ING. SANDRA VALLEJO CHAMORRO

UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA INGENIERÍA DE SISTEMAS
SAN JUAN DE PASTO
2016

NOTA DE RESPONSABILIDAD

Las ideas, recomendaciones y conclusiones planteadas en este trabajo de grado son de responsabilidad exclusiva del autor.

Artículo 1 del acuerdo No. 324 de octubre 11 de 1966, emanado del honorable Consejo Directivo de la Universidad de Nariño.

“La Universidad de Nariño no se hace responsable de las opiniones o resultados obtenidos en el presente trabajo y para su publicación priman las normas sobre derechos de autor”.

Artículo 13, Acuerdo No. 005 de 2010 emanado del Honorable Consejo Académico.

NOTA DE ACEPTACION

Firma del Director de Tesis

Jurado

Jurado

San Juan de Pasto, 22 de agosto de 2016

RESUMEN

Este trabajo fue realizado con el fin de actualizar el software Financial Business System de la Corporación Nariño Empresa y Futuro Contactar mediante su migración a la tecnología Windows Communication Foundation, se desarrolló en la sede administrativa de la organización ubicada en San Juan de Pasto.

Para llevar a cabo este proceso, inicialmente se estudió la tecnología Windows Communication Foundation (WCF) y la arquitectura del software Financial Business System, se analizaron las diferentes opciones de configuración que ofrece WCF y se definieron los cambios que se necesitaban en las diferentes soluciones de Visual Studio que componen Financial para implementar la configuración seleccionada, además para agilizar la migración, se desarrollaron programas para migrar código de forma automática donde se realizaban procesos mecánicos, posteriormente se realizaron pruebas de migración y un simulacro con código actualizado sobre un servidor de pruebas.

Por último, se realizó la actualización sobre la plataforma de producción, ejecutando los pasos especificados en cada una de las guías desarrolladas, se creó y publicó el instalador del cliente de Financial, y se confirmó el acceso y correcto funcionamiento desde varias locaciones de la organización. Adicionalmente se actualizaron las guías del proceso de desarrollo de software y de puesta a punto de los equipos de desarrollo del área de programación de Contactar.

ABSTRACT

This work was made with the objective of updating the financial business system software of the corporación nariño empresa y futuro contactar through his migration to the windows communication foundation technology.

This process was developed in the administrative office of the organization which is placed in san juan de pasto.

To develop this process, the windows communication foundation technology and the financial business system software were initially studied, the different configuration options that wcf provides were analyzed and the changes that were needed for the selected configuration in the different visual studio solutions that compound financial were defined, also to improve the migration several programs were developed to migrate code automatically in the cases where the process was mechanic, thereafter some migration test were made and a simulation with updated code over a test server was performed.

By last the update was made over the production environment, executing the steps specified in each one of the guidelines developed, the financial business system client was created and published and finally the correct access and performance across several locations of the organization were confirmed. in addition some guidelines of the software developing process and the configuration of the development computers of the programming department of contactar were updated.

CONTENIDO

	pág.
INTRODUCCIÓN	20
1. MARCO REFERENCIAL.....	25
1.1. MARCO CONTEXTUAL.....	25
1.1.1. Información general y ubicación.....	25
1.1.2. Misión.....	25
1.1.3. Visión	25
1.1.4. Objetivos	25
1.1.5. Servicios.....	26
1.1.6. Mapa de procesos.....	26
2. DESARROLLO DE LA PASANTÍA.....	27
2.1. WINDOWS COMMUNICATION FOUNDATION	27
2.2. DEFINICIÓN DE ESTRUCTURA DE FINANCIAL Y CONFIGURACIONES GENERALES CON TECNOLOGÍA WCF	33
2.3. AUTOMATIZACIÓN DE PROCEDIMIENTOS.....	43
2.3.1. Rename-It!	43
2.3.2. Macros de Visual Studio	44
2.4. MIGRACIÓN SOLUCIÓN SERVIDOR	45
2.4.1. Creación de Interfaces	45
2.4.1.1. Copiar FBSServicios	45
2.4.1.2. Renombrar archivos en lote	45
2.4.1.3. Cargar interfaces a la solución FBSServidor	48
2.4.1.4. Codificar interfaces	52
2.4.1.5. Agregar referencias.....	54

2.4.1.6.	Actualizar usings y namespace del proyecto	55
2.4.1.7.	Ajustes a archivos específicos	58
2.4.1.8	Compilar proyecto WCFInterfaces y corregir errores	58
2.4.2.	Modificar proyecto FBSServicios	59
2.4.2.1.	Formatear proyecto.....	59
2.4.2.2.	Eliminar cabeceras de servicios.....	60
2.4.2.3.	Agregar nuevas clases.....	61
2.4.2.4.	Agregar referencias.....	62
2.4.2.5.	Agregar using a WCFInterfaces.....	62
2.4.2.6	Revert a archivos específicos	63
2.4.2.7.	Correcciones manuales	64
2.4.2.8.	Quitar métodos de la interfaz	65
2.4.2.9.	MessageSettings	65
2.4.2.10.	Agregar using de WCFInterfaces.....	67
2.4.2.11.	Excluir archivos.....	68
2.4.2.12.	Agregar ErrorHandler.....	70
2.4.2.13.	Compilar proyecto	70
2.4.3.	Convertir el proyecto FBSServicios a un proyecto tipo WCF	70
2.4.3.1.	Copiar FBSServicios a un proyecto tipo WCF	70
2.4.3.2.	Agregar referencias.....	73
2.4.3.3.	Revert a AssemblyInfo.cs	76
2.4.3.4.	Editar settings de FBSServidor	77
2.4.3.5.	Ajustes manuales.....	78
2.4.3.6.	Buscar y reemplazar	78
2.4.3.7.	Compilar FBSServicios y corregir errores	79
2.4.4.	Modificar proyecto FBSMensajería	80
2.4.4.1.	Agregar usings necesarios.....	80
2.4.4.2.	Agregar librería a las referencias del proyecto.....	81
2.4.4.3.	Ejecutar macro FBSMensajería_AgregarEncabezados	82
2.4.4.4.	Compilar proyecto	82
2.4.5.	Crear Web Site	83

2.4.5.1.	Agregar Web Site a la solución.....	83
2.4.5.2.	Copiar FBSWebServices	85
2.4.5.3.	Renombrar archivos utilizando el programa Rename-it	86
2.4.5.4.	Cargar carpetas en FBSWcfServices.....	89
2.4.5.5.	Modificar código de archivos svc	90
2.4.5.6.	Agregar servicios al Web.config.....	92
2.4.5.7.	Agregar referencias.....	101
2.4.5.8.	Compilar Web Site	101
2.4.6.	Agregar y configurar aplicación en IIS	101
2.4.6.1.	Agregar aplicación FBSWcfServices.....	101
2.4.6.2.	Configurar aplicación	104
2.4.7.	Configuraciones extras	108
2.4.7.1.	Convertir XmlDocument a XElement	108
2.4.8.	Configuraciones adicionales del equipo.....	113
2.4.8.1.	Ampliar límite de tamaño del archivo web.config para IIS.....	113
2.4.8.2.	Dar permisos de acceso a la clave privada del certificado.....	117
2.4.8.3.	Habilitar servicios de net.tcp en Windows.....	119
2.5.	MIGRACIÓN SOLUCIÓN PROXIES.....	122
2.5.1.	Cambiar a Framework 4.0 los proyectos del proxy	122
2.5.2.	Obtener links actualizados de los Service References	123
2.5.3.	Importar y configurar librería WCFExtras.....	127
2.5.4.	Ejecutar macro de actualización de service references	129
2.6.	MIGRACIÓN SOLUCIÓN CLIENTE.....	130
2.6.1.	Actualizar todos los proyectos a Framework 4.0.....	130
2.6.2.	Ejecutar macro de actualización de proyecto FBSProxies.....	131
2.6.3.	Referencias.....	133
2.6.4.	Excluir archivos de la solución	134
2.6.5.	Actualizar excepciones	134
2.6.6.	Buscar y reemplazar	137
2.6.7.	Ajustes finales.....	138

2.6.8.	Compilar y corregir errores en caso de ser necesario	141
3.	CONCLUSIONES	142
4.	RECOMENDACIONES	143
	REFERENCIAS BIBLIOGRÁFICAS.....	144

LISTA DE TABLAS

	pág.
Tabla 1 WCFInterfaces - Archivos excluidos	51
Tabla 2 FBSServicios - Archivos a excluir del proyecto	75
Tabla 3 FBSServicios - Relación buscar y reemplazar	75

LISTA DE FIGURAS

	pág.
Figura 1	Mapa de procesos de Contactar.....26
Figura 2	Arquitectura de WCF30
Figura 3	Elección de transporte34
Figura 4	Transportes y codificaciones para bindings comunes.....35
Figura 5	Elección de hosting.....36
Figura 6	Estructura solución FBSServidor38
Figura 7	Composición de solución FBSServidor.....38
Figura 8	Estructura solución FBSProxies39
Figura 9	Composición solución FBSProxies40
Figura 10	Estructura solución FBSCliente41
Figura 11	Composición solución FBSCliente.....42
Figura 12	Elementos innecesarios para renombrar45
Figura 13	Rename-It - Elección de filtro.....46
Figura 14	Rename-It - Expresión de reemplazo46
Figura 15	Rename-It - Elección de carpetas.....46
Figura 16	Rename-It - Elección de carpetas 2.....47
Figura 17	Rename-It - Resumen de configuración47
Figura 18	Rename-It - Proceso terminado.....48
Figura 19	Rename-It - Archivos renombrados48
Figura 20	WCFInterfaces - Crear proyecto49
Figura 21	WCFInterfaces - Archivos automáticos a eliminar49
Figura 22	WCFInterfaces - Mover archivos desde temp.....50
Figura 23	WCFInterfaces - Incluir archivos en proyecto50
Figura 24	WCFInterfaces - Archivos incluidos51
Figura 25	Macro V4_WCFInterfaces - ModuloRecorrerProyecto.....52
Figura 26	V4_WCFInterfaces - RealizarProcesamientoCompleto53
Figura 27	V4_WCFInterfaces - Subproyectos procesados53
Figura 28	Archivos MessageSettings.cs y RaiseException.cs54
Figura 29	WCFInterfacse - Agregar DLL de WCFExtras55
Figura 30	WCFInterfaces - Actualizar nombre de namespace.....56
Figura 31	WCFInterfaces - Namespace modificado56
Figura 32	WCFInterfaces - Eliminar importación de FBSServicios57
Figura 33	WCFInterfaces - Importación de FBSServicios eliminada57
Figura 34	WCFInterfaces - Eliminar cabeceras de SeguridadesFBS/ILoginWS ..58
Figura 35	WCFInterfaces - Eliminar declaración de objeto valoresACobrar58
Figura 36	WCFInterfaces - Posible error por no importación de SoapHeaders ...58
Figura 37	FBSServicios - Macro FormatearArchivosProyecto59

Figura 38	FBSServicios - FormatearArchivosProyecto - ProcesarProyecto	60
Figura 39	FBSServicios - Ejecutar macro V4_FBSServicios	61
Figura 40	FBSServicios - Incluir clases desde recursos	61
Figura 41	FBSServicios - Agregar using de System.ServiceModel	62
Figura 42	FBSServicios - Agregar importación de WCFInterfaces	63
Figura 43	FBSServicios - Revert a referencia web de data crédito.....	64
Figura 44	FBSServicios - Agregar endregion a Personas.RegimenSaludWS.cs.	64
Figura 45	FBSServicios - Ejemplo método que no es servicio web	65
Figura 46	FBSServicios - Agregar variable MessageSettings.....	66
Figura 47	FBSServicios - Actualizar instanciación de variable MessageSettings	66
Figura 48	FBSServicios - Excluir archivo MessageSettings.....	67
Figura 49	FBSServicios - Modificar importación de WCFInterfaces	68
Figura 50	FBSServicios - Excluir archivo RaiseException	69
Figura 51	FBSServicios - Compilar proyecto	70
Figura 52	FBSServicios - Agregar nuevo proyecto	71
Figura 53	FBSServicios - Elegir tipo de proyecto.....	71
Figura 54	FBSServicios - Eliminar archivos autogenerados	72
Figura 55	FBSServicios - Activar opción mirar todos los archivos	72
Figura 56	FBSServicios - Incluir carpetas y archivos en nuevo proyecto	73
Figura 57	FBSServicios - Ejemplo de búsqueda y reemplazo	76
Figura 58	FBSServicios - Revert a AssemblyInfo.cs.....	76
Figura 59	FBSServicios - Modificar Settings.....	77
Figura 60	FBSServicios - Agregar setting.....	77
Figura 61	FBSServicios - Comentar cabeceras de SeguridadesFBS/MenuWS ..	78
Figura 62	FBSServicios - Eliminar System.Web.HttpContext.Current.User.....	79
Figura 63	FBSServicios - Compilar proyecto	80
Figura 64	FBSMensajería - Importar System.Runtime.Serialization	81
Figura 65	FBSMensajería - Agregar librería System.Runtime.Serialization.....	81
Figura 66	FBSMensajería - Ejecución FBSMensajería_AgregarEncabezados ...	82
Figura 67	FBSMensajería - Compilar proyecto	83
Figura 68	Web Site - Nuevo.....	84
Figura 69	Web Site - Ruta de creación	84
Figura 70	Web Site - Creación terminada.....	85
Figura 71	Web Site - Agregar carpetas para renombrar	86
Figura 72	Web Site - Carpeta temporal	86
Figura 73	Web Site - Selección de parte para renombrar	87
Figura 74	Web Site - Agregar filtro Seach & Replace	87
Figura 75	Web Site - Ingreso de la nueva extensión	88
Figura 76	Web Site - Iniciar procesamiento	88
Figura 77	Web Site - Renombrado terminado.....	89
Figura 78	Web Site - Resultado de renombramiento.....	89
Figura 79	Web Site - Copiar carpetas renombradas.....	89
Figura 80	Web Site - Estructura con las nuevas carpetas	90
Figura 81	Web Site - Modificar código de archivos svc	91

Figura 82	Web Site - Confirmar reemplazo.....	91
Figura 83	Web Site - Reemplazo terminado	91
Figura 84	Web Site - Ejemplo código de archivo svc.....	92
Figura 85	Web Site - Buscar en todo el proyecto.....	92
Figura 86	Web Site - Resultado de búsqueda	93
Figura 87	Excel - Resultado en excel	93
Figura 88	Excel - Duplicados eliminados	94
Figura 89	Excel - Seleccionar todos los datos	94
Figura 90	Excel - Datos como texto en columnas.....	95
Figura 91	Excel - Elegir tipo de archivo	95
Figura 92	Excel - Elegir caracter separador.....	96
Figura 93	Excel - Elegir formado de las columnas.....	96
Figura 94	Excel - Borrado de columnas.....	97
Figura 95	Excel - Eliminar carácter especial.....	97
Figura 96	Excel - Eliminación terminada.....	97
Figura 97	Excel - Eliminar carácter XML.....	98
Figura 98	Excel - Carácter XML eliminado	98
Figura 99	Excel - Verificación	98
Figura 100	Web.config - Backup del Web.config	99
Figura 101	Web.config - Reemplazar contenido del Web.config	99
Figura 102	Web.config - Reemplazo con expresión regular	100
Figura 103	Web.config - Reemplazo terminado.....	100
Figura 104	Web.config - Ejemplo resultado etiqueta Service con contenido	101
Figura 105	IIS - Agregar aplicación.....	101
Figura 106	IIS - Ingresar alias de la aplicación	102
Figura 107	IIS - Seleccionar usuario de conexión.....	102
Figura 108	IIS - Prueba de conexión	103
Figura 109	IIS - Nueva aplicación creada	103
Figura 110	IIS - Configuración avanzada de aplicación.....	104
Figura 111	IIS - Estado inicial de configuración.....	104
Figura 112	IIS - Protocolo net.tcp agregado a la aplicación.....	105
Figura 113	IIS - Abrir administrar sitio web	105
Figura 114	IIS - Configuración original del web site.....	106
Figura 115	IIS - Protocolo net.tcp agregado al sitio web.....	106
Figura 116	IIS - Modificar enlaces	107
Figura 117	IIS - Enlaces de sitios	107
Figura 118	Configuraciones extras - Actualizar XmlDocument 1	108
Figura 119	Configuraciones extras - Convertir a XmlElement 1	108
Figura 120	Configuraciones extras - Actualización de interfaz 1	109
Figura 121	Configuraciones extras - Verificación de acceso 1	109
Figura 122	Configuraciones extras – Actualizar XmlDocument 2	110
Figura 123	Configuraciones extras – Convertir a XmlElement 2	110
Figura 124	Configuraciones extras - Actualizar interfaz 2.....	111
Figura 125	Configuraciones extras - Actualizar XmlDocument 3.....	111

Figura 126 Configuraciones extras - Modificar XmlElement 3	112
Figura 127 Configuraciones extras - Actualizar interfaz 3.....	112
Figura 128 Configuraciones extras – Verificación de acceso 3	112
Figura 129 Regedit - Iniciar funcionalidad Editor del registro.....	113
Figura 130 Regedit - Editor del registro iniciado	113
Figura 131 Regedit - Nueva clave.....	114
Figura 132 Regedit - Clave Configuration creada.....	114
Figura 133 Regedit - Nuevo valor DWORD	115
Figura 134 Regedit – Ingresar nombre de valor	115
Figura 135 Regedit - Modificar valor de la clave	116
Figura 136 Regedit - Ingresar valor de la clave	116
Figura 137 Regedit - Valor creado	116
Figura 138 IIS - Reiniciar servicio del IIS	117
Figura 139 IIS - Reinicio de IIS en proceso	117
Figura 140 MMC - Iniciar la consola de administración de Microsoft.....	117
Figura 141 MMC - Verificar existencia del certificado WcfFinancial.....	118
Figura 142 MMC - Administrar claves privadas del certificado	118
Figura 143 MMC - Asignar permiso de lectura a usuario de IIS	119
Figura 144 WCF - Iniciar funcionalidad de Características de Windows.....	119
Figura 145 WCF - Activar características de WCF	120
Figura 146 WCF - Iniciar funcionalidad Servicios	120
Figura 147 WCF - Configurar inicio de servicios de Net.Tcp	121
Figura 148 WCF - Iniciar consola de Visual Studio.....	121
Figura 149 WCF - Ejecutar comando aspnet_regiis.exe -iru	121
Figura 150 WCF - Reiniciar IIS	122
Figura 151 FBSProxies - Actualizar versión de .NET Framework.....	122
Figura 152 FBSProxies - .NET Framework actualizado.....	123
Figura 153 FBSProxies - Buscar links de servicios.....	123
Figura 154 FBSProxies - Resultado de la búsqueda	124
Figura 155 FBSProxies - Ajustar extensión de filas	124
Figura 156 FBSProxies - Eliminar sección inicial de cada fila	125
Figura 157 FBSProxies - Insertar url de conexión	125
Figura 158 FBSProxies - Reemplazar extensión de los links	126
Figura 159 FBSProxies - Listado de links para las Service References	126
Figura 160 FBSProxies - Insertar código XML para usar librería WCFExtras	127
Figura 161 FBSProxies - Agregar DLL de WCFExtras	128
Figura 162 FBSProxies - Ejemplo etiqueta metadata para el app.config.....	129
Figura 163 FBSProxies - Eliminar carpetas excluidas del proyecto.....	129
Figura 164 FBSProxies - Macro FBSProxies_ActualizarWebReferences.....	130
Figura 165 FBSCliente - Actualizar proyectos a .NET Framework 4.0	130
Figura 166 FBSCliente - Macro FBSCliente_ModificarProxy	131
Figura 167 FBSCliente - Datos de macro FBSCliente_ModificarProxy.....	131
Figura 168 FBSCliente - Agregar carpeta WcfComun	132
Figura 169 FBSCliente - Carpeta WcfComun creada	132

Figura 170 FBSCliente - Importar clases a WcfComun	132
Figura 171 FBSCliente - Clases importadas a WcfComun	133
Figura 172 FBSCliente - Agregar using para comunicación con WCF	134
Figura 173 FBSCliente - Sentencias de nuevos using agregadas	134
Figura 174 FBSCliente - Actualizar tipo de excepción	135
Figura 175 FBSCliente - Actualizar código de ProcesaSoapException	135
Figura 176 FBSCliente - Actualizar llamadas a excepciones.....	136
Figura 177 FBSCliente - Actualizar verificación de tipos de excepción	137
Figura 178 FBSCliente - Actualizar using de Activo_Baja	137
Figura 179 FBSCliente - Buscar y reemplazar.....	138
Figura 180 FBSCliente - Agregar nuevas settings	138
Figura 181 FBSCliente - Actualizar código Devuelve() de LoginProxy.cs.....	139
Figura 182 FBSCliente - Actualizar UsuarioComplementoProxy	141
Figura 183 FBSCliente - Actualizar LoginForm1.cs	141

GLOSARIO

API: interfaz de Programación de Aplicaciones, es el conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

APLICACIÓN WEB: en la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.

BINDING: define como un endpoint se comunica con el mundo. Está conformado por un conjunto de componentes llamados elementos de binding apilados uno sobre otro para crear la infraestructura de comunicación. Al final el binding define el transporte (tal como HTTP o TCP) y la codificación usada (tal como texto o binario). Un binding puede contener elementos de binding que especifican detalles tales como los mecanismos de seguridad usados para proteger mensajes, o el patrón de mensajes usados por un endpoint.

CLUSTER: el término clúster (del inglés cluster, "grupo" o "raíz") se aplica a los conjuntos o conglomerados de ordenadores unidos entre sí normalmente por una red de alta velocidad y que se comportan como si fuesen una única computadora.

COM+: es una evolución de COM (Microsoft Component Object Model) y MTS (Microsoft Transaction Server). COM+ incorpora la asignación de hilos y seguridad. Provee una mayor escalabilidad que sus antecesores con la incorporación de la agrupación de hilos, agrupación de objetos y la activación de objetos just-in-time (justo a tiempo).

DCOM: el Modelo de Objetos de Componentes Distribuidos (Distributed Component Object Model, DCOM) es una tecnología propietaria de Microsoft para desarrollar componentes de software distribuidos sobre varias computadoras y que se comunican entre sí.

ENDPOINT: extremo expuesto por la aplicación que corresponde a un contrato de servicios implementado por la aplicación, donde se exponen una o más operaciones del servicio.

FRAMEWORK: es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras

herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

HOST: el término host ("anfitrión", en español) es usado en informática para referirse a las computadoras conectadas a una red, que proveen y utilizan servicios de ella. Los usuarios deben utilizar anfitriones para tener acceso a la red. En general, los anfitriones son computadores monousuario o multiusuario que ofrecen servicios de transferencia de archivos, conexión remota, servidores de base de datos, servidores web, etc.

IDE: ambiente de Desarrollo Integrado (Integrated Development Environment) es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

JSON: acrónimo de JavaScript Object Notation, es un formato de texto ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript aunque hoy, debido a su amplia adopción como alternativa a XML, se considera un formato de lenguaje independiente.

MTOM: es el Mecanismo de Optimización de Transmisión de Mensajes (Message Transmission Optimization Mechanism) de la W3C, un método para enviar eficientemente datos binarios desde y hacia servicios web.

RIA: una Rich Internet Application (RIA), "aplicación de Internet enriquecida" o "aplicación rica de internet" (ARI), es una aplicación web que tiene la mayoría de las características de las aplicaciones de escritorio tradicionales. Estas aplicaciones utilizan un navegador web estandarizado para ejecutarse y por medio de complementos o mediante una máquina virtual se agregan las características adicionales.

SERVICIO WEB: (en inglés, Web Service o Web services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet.

SOA: la Arquitectura Orientada a Servicios (siglas del inglés Service Oriented Architecture) es un paradigma de arquitectura para diseñar y desarrollar sistemas distribuidos. Las soluciones SOA han sido creadas para satisfacer los objetivos de negocio las cuales incluyen facilidad y flexibilidad de integración con sistemas legados, alineación directa a los procesos de negocio reduciendo costos de implementación, innovación de servicios a clientes y una adaptación ágil ante cambios incluyendo reacción temprana ante la competitividad.

SOAP: (siglas de Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por Dave Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros. Está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

VISUAL STUDIO: es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta múltiples lenguajes de programación tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby, PHP; al igual que entornos de desarrollo web como ASP.NET MVC, Django, etc., a lo cual sumarle las nuevas capacidades online bajo Windows Azure en forma del editor Monaco.

W3C: el Consorcio World Wide Web (W3C) es una comunidad internacional donde las organizaciones Miembro, personal a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares Web. Liderado por el inventor de la Web Tim Berners-Lee y el Director Ejecutivo (CEO) Jeffrey Jaffe, la misión del W3C es guiar la Web hacia su máximo potencial.

WCF: Windows Communication Foundation es un modelo de programación unificado para construir aplicaciones orientadas a servicios.

WS-*: abreviatura para el conjunto de especificaciones de Servicios Web (WS), tales como WS-Security, WS-ReliableMessaging y demás, que son implementadas en WCF.

WSDL: son las siglas de Web Services Description Language, un formato XML que se utiliza para describir servicios Web. WSDL describe la interfaz pública a los servicios Web. Está basado en XML y describe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligan después al protocolo concreto de red y al formato del mensaje.

WSE: Web Services Enhancements es un complemento actualmente obsoleto de Microsoft .NET Framework, el cual provee un conjunto de herramientas que los desarrolladores usan para asegurar servicios web interoperables (ASMX) basados en varias de las especificaciones WS-*, tales como WS-Security, WS-Addressing y WS-Policy.

INTRODUCCIÓN

Actualmente un cúmulo de organizaciones, de pequeña, mediana y gran escala adquieren herramientas tecnológicas con el objeto de obtener una mayor competitividad en el mercado en el que se desenvuelven, apoyadas en criterios de elección que se originan en intereses que están directamente relacionados con el modelo de negocio de cada organización.

Para el caso concreto, la Corporación Nariño Empresa y Futuro Contactar, ha adquirido en el transcurso de varios años de operación, distintos software que permiten la gestión de los procesos que existen en las diferentes áreas de la empresa. Una de las herramientas más importantes que posee es Financial Business System, software por medio del cual se gestionan las actividades relacionadas con el otorgamiento de créditos, el seguimiento a la cartera y las operaciones concernientes al ingreso y egreso en las bóvedas y cajas, entre otras operaciones.

Si bien es cierto que Financial Business System es en la actualidad un software considerado como robusto por su estabilidad y la gran cantidad de funcionalidades que provee, la tecnología con la que implementa sus servicios web (Web Services Enhancements) no es la mejor del momento, pues existen otro tipo de tecnologías que brindan mayor beneficio y mejor desempeño en el desarrollo de software orientado a servicios, motivo por el cual, WSE ha dejado de ser la opción preferida de la plataforma .NET.

Teniendo en cuenta lo anterior, este trabajo se orientó a la actualización de la tecnología de comunicación de Financial Business System, a través de la migración total del software a Windows Communication Foundation, manteniendo la base de los estándares de programación y adaptando el proceso de desarrollo software del área de programación de la organización.

CONTEXTUALIZACIÓN DE LA PASANTÍA

DESCRIPCIÓN DEL PROBLEMA

Cada día las organizaciones reconocen más la importancia de los sistemas de información dentro de su entorno, todos los elementos que lo conforman (software, hardware, bases de datos, personas y procesos) se han convertido en un activo de gran relevancia para todo tipo de compañía brindando una ventaja o desventaja respecto a sus competidores. Esta situación ha promovido la realización de considerables esfuerzos e inversiones en pro de la implementación de sistemas de información que mejoren la productividad de las empresas.

Dentro de un Sistema de Información, el software toma un lugar importante debido al gran impacto que puede significar su correcto funcionamiento y adopción por parte de los usuarios, además existen diversos criterios que marcan la percepción que posee el usuario acerca del software. Estos criterios pueden ser: facilidad de uso, seguridad, confiabilidad, velocidad de respuesta, entre otros.

Para mejorar cada uno de estos criterios se puede optar por diferentes alternativas como mejorar la infraestructura, la red de comunicaciones, adquirir equipos de cómputo con mejor rendimiento o implementar las tecnologías más recientes en el campo del software.

Actualmente el software transaccional de Contactar es Financial Business System, este implementa sus servicios web bajo WSE 3.0 (Web Services Enhancements), el cual es un complemento de Microsoft .NET Framework que se considera actualmente como una herramienta obsoleta para la implementación de servicios web.

Debido al continuo crecimiento en la organización y a la exigencia de un desempeño adecuado por parte de Financial Business System, en Contactar se ha optado por actualizar la tecnología de este software para que sea implementado utilizando la tecnología WCF, aprovechando así los más recientes avances que Microsoft ha desarrollado en el mundo de los servicios web, tales como seguridad, rendimiento, flexibilidad, estabilidad, manejo de transacciones y concurrencia, entre otros.

FORMULACIÓN DEL PROBLEMA

¿Cómo incorporar al software Financial Business System de la Corporación Nariño Empresa y Futuro Contactar los avances en aplicaciones orientadas a servicios más recientes de la plataforma .NET?

OBJETIVOS

Objetivo general

Actualizar el software Financial Business System de la Corporación Nariño Empresa y Futuro Contactar a través de su migración a la tecnología Windows Communication Foundation.

Objetivos específicos

- Identificar y conocer las características de WCF para determinar los cambios que se deben realizar a la estructura y al código de Financial Business System.
- Analizar y definir las configuraciones de WCF para servidor y cliente que mejor se adapten a las necesidades de la organización.
- Desarrollar funcionalidades que automaticen los procesos mecánicos necesarios para migrar el código de la solución servidor y de la solución cliente.
- Definir los procedimientos necesarios para migrar Financial Business System, con el fin de disminuir los inconvenientes al momento de ejecutar la migración sobre el entorno de producción.

JUSTIFICACIÓN

Corporación Nariño Empresa y Futuro - Contactar es una empresa que presta servicios microfinancieros a poblaciones prioritariamente rurales, la cual tiene su sede administrativa ubicada en San Juan de Pasto y además tiene oficinas en los departamentos de Nariño, Putumayo, Huila y Tolima. Para administrar la información de las áreas existentes como el área contable, financiera, comercial y de nómina, entre otras, Contactar compró el software Financial Business System V2 a la empresa ecuatoriana SifizSoft, incluyendo en la compra la adquisición del código fuente en su totalidad; este hecho permitió la creación de un grupo de

desarrollo de software interno enfocado en acondicionar las funcionalidades existentes e implementar nuevas características que se adaptaran a las necesidades específicas de la organización.

El desarrollo de Financial ha sido constante durante los últimos años manteniendo sus estándares de programación originales, convirtiendo a Financial en un software cada vez más robusto y con mayor cantidad de funcionalidades. Las características tecnológicas relevantes que destaca Sifizsoft sobre Financial son:

- Desarrollado en Visual Studio .NET 2010, la más moderna herramienta de desarrollo producida por Microsoft.
- Sistema Multi-Capas, garantizando una total escalabilidad.
- Código fuente C# (Sharp), el lenguaje de desarrollo de mayor difusión y versatilidad en la actualidad.
- Multi - Plataforma Base De Datos, Oracle 10g / 11 g, SQL Server 2005 / 2008, IBM DB2, IBM Informix, Sybase.
- Arquitectura Orientada a Servicios (SOA), alta cohesión, escalabilidad y acoplamiento.
- Tecnología Smart Client, Front End y Back End Windows (le proporciona interacción con el Usuario de Gran Calidad).
- Web Services XML (Máximo Rendimiento y Escalabilidad).
- Fácil Integración con tecnologías y canales de acceso a terceros.
- Encriptamiento y Compresión de mensajes proporcionando total seguridad y alto rendimiento.
- Estaciones de Trabajo Windows XP, 2000, Vista, Windows 7 y Windows 8.
- Autogeneración de código, simplificando el trabajo de los desarrolladores y minimizando los errores de sintaxis y estandarización. (SifizSoft S.A., 2015)

Aunque la mayoría de las características mencionadas aún se consideran acertadas, la relacionada con los Web Services hace varios años ha pasado a ser errónea debido a que la tecnología de servicios web que utiliza Financial llamada WSE (Web Services Enhancements) se considera como obsoleta desde la aparición y posterior acogida de la tecnología WCF (Windows Communication Foundation) como parte de .NET en 2005, año de lanzamiento de la primera versión de lo que se considera ahora como el modelo de programación unificado de Microsoft para construir aplicaciones orientadas a servicios. WCF permite a los

desarrolladores construir soluciones seguras, confiables y transaccionales, que se integran entre plataformas y que interoperan con inversiones existentes. Además posee varias mejoras con respecto a WSE 3.0, entre las que se incluyen la mejoría en la velocidad de respuesta, interoperabilidad con sistemas que no pertenecen a .NET, inclusión de tipos de codificación de mejor rendimiento, mayor cantidad de mecanismos de transporte, de serialización y seguridad mejorada.

Todos estos aspectos permiten concluir que la adopción de WCF permitirá a Financial obtener los siguientes beneficios: (Advantages of WCF over Web Services, 2013)

- Reducción del tráfico de red.
- Mejor velocidad de respuesta.
- Mejores estándares de seguridad.
- Soporte para una mayor variedad de protocolos.
- Mayor variedad en las opciones de hospedaje de los servicios web.
- Facilidad para la integración con otras plataformas.

1. MARCO REFERENCIAL

1.1. MARCO CONTEXTUAL

1.1.1. Información general y ubicación. La Corporación Nariño Empresa y Futuro Contactar es una empresa que presta servicios microfinancieros en Nariño, Putumayo, Tolima y Huila (Colombia); sus datos son los siguientes:

- Dirección sede administrativa: Calle 17 No. 29-31
- Teléfono: 7316693
- E-mail: info@contactar-pasto.org

1.1.2. Misión. Prestamos servicios microfinancieros inclusivos e integrales con calidad, calidez, responsabilidad social y ambiental, a poblaciones prioritariamente rurales, para contribuir a mejorar sus condiciones de vida¹.

1.1.3. Visión. Contactar será modelo empresarial de microfinanzas prioritariamente rurales con desempeño social y ambiental en Colombia¹.

1.1.4. Objetivos. Los objetivos de Contactar son los siguientes¹:

- Prestar servicios microfinancieros inclusivos e integrales a poblaciones prioritariamente rurales.
- Garantizar el cumplimiento de los tiempos de atención en términos competitivos.
- Mantener tasas y descuentos competitivos.

¹ CONTACTAR. Manual de calidad. 2016. Disponible en Internet: <http://www.isolucion.co/isolucion3contactar> [Citado el 22-Ago-2016]

- Mantener un alto nivel de satisfacción del cliente.
- Garantizar personal competente y comprometido.
- Cumplir las metas organizacionales, garantizando la gestión eficiente de los recursos.
- Implementar, mantener y mejorar continuamente los procesos del sistema de gestión de calidad.

1.1.5. Servicios. La Corporación Nariño Empresa y Futuro Contactar ofrece los siguientes servicios de microcrédito:

- Crédito individual.
- Crédito grupal.
- Crédito con su planeta.
- Crédito adicional.

1.1.6. Mapa de procesos. El mapa de procesos de la Corporación Nariño Empresa y Futuro Contactar está conformado de la siguiente manera:



Figura 1 Mapa de procesos de Contactar

2. DESARROLLO DE LA PASANTÍA

Este informe documenta el proceso de actualización de tecnología del software Financial Business System mediante su migración a WCF. Inicialmente se recopilan las principales características de WCF, luego se realiza la selección de las principales configuraciones de acuerdo al contexto de la Corporación Nariño Empresa y Futuro Contactar, se documenta el proceso realizado para adaptar la estructura de las soluciones servidor, proxy y cliente de Visual Studio que componen Financial, luego se aborda la creación de funcionalidades enfocadas en la automatización de la migración de código en las diferentes soluciones. Posteriormente se detalla el simulacro realizado con el grupo de programación, y finalmente se realizan los ajustes necesarios a la guía de migración y a los archivos anexos, lo cual permite realizar la actualización a WCF en ambiente de producción.

La metodología aplicada en el desarrollo de estas actividades obedeció a diseñar la estructura de las soluciones servidor y cliente para ajustarse a la arquitectura de WCF, desarrollar programas para automatizar las operaciones de reescritura de código y elegir las configuraciones que se ajustaran a las necesidades de la organización.

2.1. WINDOWS COMMUNICATION FOUNDATION

Windows Communication Foundation (WCF) es un marco de trabajo para desarrollar y desplegar aplicaciones orientadas a servicios en Windows. Usando WCF se pueden enviar datos como mensajes asíncronos desde un extremo de servicio a otro. Un extremo de servicio puede ser parte de un servicio continuamente disponible hospedado en IIS, o puede ser hospedado en una aplicación. Un extremo de servicio puede ser un cliente u otro servicio que realice peticiones de datos. El mensaje puede ser tan sencillo como un simple caracter o palabra enviada como XML, o tan complejo como un flujo de datos binarios.

WCF está basado en la noción de comunicación basada en mensajes. Los mensajes son enviados entre Endpoints. Endpoints son los lugares donde los mensajes son enviados o recibidos (o ambos), y definen toda la información requerida para el intercambio de mensajes. Un servicio expone uno o más

endpoints de aplicación y el cliente genera un endpoint que es compatible con uno de los endpoints del servicio. Un endpoint describe en una forma estándar a donde se deberían enviar los mensajes, como se deben enviar y como deben estar formados².

Productividad de desarrollador

WCF aumenta la productividad del desarrollador simplificando el desarrollo de aplicaciones orientadas a servicios. Previamente, los desarrolladores eran forzados a aprender diferentes APIs para construir componentes distribuidos. No se puede negar que los desarrolladores que son buenos para construir componentes de servicios tal vez no sean tan eficientes construyendo componentes remotos usando .NET Remoting. Crear soluciones personalizadas que requieren la funcionalidad de dos o más tecnologías distribuidas siempre han generado molestias para los desarrolladores y arquitectos.

WCF tiene un simple, modelo de programación unificado que ofrece el potencial de crear aplicaciones con diversos requerimientos. WCF se construyó desde la base para soportar las características de la orientación a servicios. Uno de los mejores aspectos de WCF es que los desarrolladores que usan tecnologías existentes encontrarán sus características favoritas en ella, y todos los desarrolladores se beneficiarán de su arquitectura consistente. El soporte de WCF para los modelos de programación declarativa e imperativa genera la escritura de menos código, lo cual disminuirá la probabilidad de errores. Las aplicaciones que requieran cientos o miles de líneas de código antes de WCF ahora pueden ser creadas con pocas líneas³.

WAS ofrece varias ventajas sobre self-hosting, incluyendo application pooling, reciclaje, gestión del tiempo de inactividad, gestión de identidad, aislamiento, y es el proceso de host de elección cuando está disponible, eso es cuando se despliega en una plataforma que lo soporta, tal como Windows Server 2008 (o posterior) para escalabilidad, o Windows Vista o Windows 7 (o posterior) para los clientes.

² Díaz, M., Garrido, D., & Troya, J. (s.f.). *Developing a communications architecture based on WCF for use in nuclear power plants simulators*. Universidad de Málaga, Málaga.

³ Pathak, N. (2011). *Pro WCF 4 Practical Microsoft SOA Implementation*. Apress.

Arquitectura de WCF

WCF ofrece un apoyo inmensamente valioso para fiabilidad, transacciones, gestión de concurrencia, seguridad, y activación de instancia, todos los cuales recaen en la arquitectura basada en intercepción de WCF. Hacer interactuar al cliente con el proxy significa que WCF está siempre presente entre el servicio y el cliente, interceptando la llamada y realizando procesamiento pre y post llamada. La interceptación inicia cuando el proxy serializa el marco de pila de llamada a un mensaje y envía los mensajes por una cadena de canales. El canal es meramente un interceptor cuyo propósito es realizar una tarea específica. Cada canal del lado del cliente realiza procesamiento pre llamada del mensaje. La estructura exacta y composición de la cadena depende en su mayoría del binding. Por ejemplo, uno de los canales puede ser responsable de codificar el mensaje (binario, texto o MTOM), otro de pasar la seguridad del contexto de llamada, otro de propagar la transacción del cliente, otro de gestionar la sesión confiable, otro de encriptar el cuerpo del mensaje (en caso de estar configurado), etc. El último canal en el lado del cliente es el canal de transporte, el cual envía el mensaje sobre el transporte configurado hacia el servidor.

En el lado del servidor, el mensaje pasa a través de otra cadena de canales que realizan procesamiento pre llamada del mensaje. El primer canal en el lado del servidor es el canal de transporte, el cual recibe el mensaje del transporte. Los canales subsecuentes realizan varias tareas tales como el descifrado del cuerpo del mensaje, decodificación del mensaje, unión de la transacción propagada, configurar la entidad de seguridad, gestionar la sesión, y activar la instancia del servicio. El último canal en el servidor pasa el mensaje al despachador (Dispatcher). El despachador convierte el mensaje a un marco de pila y llama a la instancia del servicio. Esta secuencia está representada en la siguiente figura⁴.

⁴ Lowy, J., & Michael, M. (2015). *Programming WCF Services (4 ed.)*. O'Reilly Media.

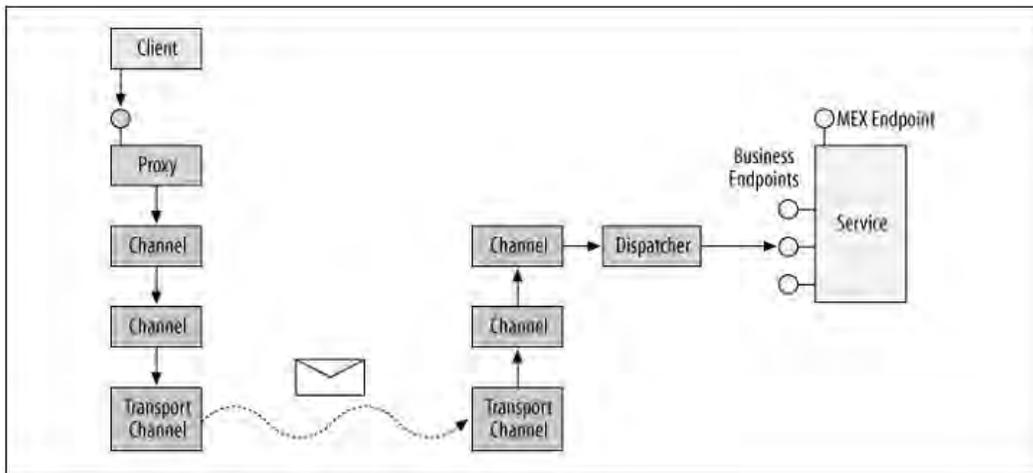


Figura 2 Arquitectura de WCF

Se puede usar WCF para crear nuevos servicios, así como aumentar e interoperar con muchos de los servicios existentes creados usando otras tecnologías. Cuando se diseñaban aplicaciones distribuidas en el pasado, frecuentemente se tenía que elegir una tecnología específica, tal como Web Services, COM+, Microsoft Message Queue, o .NET Framework Remoting. Esa elección frecuentemente tenía un impacto fundamental en la arquitectura de la solución. En contraste, WCF provee un modelo consistente para implementar sistemas escalables y extensibles que empleen una variedad de tecnologías con las cuales se pueden diseñar y establecer la arquitectura a la solución sin ser restringido por un mecanismo de conectividad específico⁵.

Antecedentes

Tecnologías tales como COM, DCOM, COM+, Enterprise Services, y .NET Framework Remoting funcionan bien cuando las aplicaciones y los componentes se están ejecutando en la misma red local dentro de una organización. Estas también son específicas para los sistemas operativos de la familia de Microsoft Windows.

Mientras Microsoft desarrollaba COM y DCOM la World Wide Web (WWW) aparecía. La www está basada en Internet, la cual ha estado alrededor hace varias décadas. La www provee una infraestructura con la cual los desarrolladores pueden construir aplicaciones que pueden combinar componentes y otros

⁵ Sharp, J. (2010). *Windows Communication Foundation 4 Step by Step*. Microsoft.

elementos localizados en casi cualquier lugar del mundo, ejecutándose en computadores de arquitecturas variadas, y que se ejecutan en una desconcertante variedad de sistemas operativos (no solamente Windows). La primera generación de “aplicaciones web” era bastante simple y consistía de páginas Web estáticas que los usuarios podían descargar y mirar usando un navegador Web ejecutándose en su computadora local. La segunda generación proveyó elementos de programación, inicialmente a través del uso de componentes, o applets, que podían ser descargados de sitios Web y ejecutarse localmente en el navegador Web del usuario. Estos han sido seguidos por la tercera generación- Servicios Web. Un Servicio Web es una aplicación o componente que se ejecuta en el computador que hospeda el sitio Web en lugar del computador del usuario. Un Servicio Web puede recibir peticiones desde aplicaciones corriendo en el computador del usuario, realizar operaciones en el computador que hospeda el servicio Web, y enviar una respuesta de vuelta a la aplicación que corre en el computador del usuario. Un servicio Web también puede invocar operaciones en otros servicios Web, hospedados en otro lugar en Internet. Estas son aplicaciones distribuidas, globales.

Se pueden crear servicios Web que se ejecuten en Windows al usar Visual Studio y el Framework .NET. Se pueden crear servicios Web para otras plataformas haciendo uso de otras tecnologías, como Java y el Java Web Services Developers Pack. Sin embargo, los servicios Web no son específicos a algún lenguaje o sistema operativo particular. Para establecer los servicios Web como un mecanismo global para construir aplicaciones distribuidas, los desarrolladores tuvieron que acordar en varios puntos, incluyendo un formato común para datos, un protocolo para enviar y recibir peticiones, y manejo de seguridad. Todas estas características tenían que ser independientes de la plataforma usada para crear y hospedar los servicios Web.

Entre los estándares más importantes que se definieron se encuentran:

- Usar XML como el formato de datos común.
- Envío y recepción peticiones de servicios web. Los servicios Web y las aplicaciones cliente se comunican usando SOAP. La especificación SOAP define un número de aspectos, de los cuales los más importantes son:
 - El formato del mensaje SOAP.
 - Cómo los datos deben ser codificados.

- Cómo enviar los mensajes.
- Cómo manejar las respuestas a estos mensajes⁶.
- Seguridad y privacidad en un ambiente global: Enfocada en identificar usuarios, servicios y autorizar su acceso a recursos. OASIS (Organización para el Avance de Estándares de Información Estructurada) ha propuesto un número de mecanismos estándar para la implementación de seguridad, tal como el uso de pares usuario/contraseña, certificados X509, y tokens Kerberos. Al crear servicios web que provean acceso a información privilegiada, se debería considerar uno de estos mecanismos para autenticar usuarios.

Un servicio Web puede publicar un documento WSDL (Web Services Description Language), el cual es un fragmento XML que describe el mensaje que el servicio Web puede aceptar y la estructura de las respuestas que enviará de vuelta. Una aplicación cliente puede usar esta información para determinar cómo se comunicará con el servicio Web.

- JSON y RIA
- Manejo de la seguridad y privacidad en un ambiente global

Arquitectura orientada a Servicios y Windows Communication Foundation

Los desarrolladores de software pronto apreciaron que los principios de los servicios Web podrían ser aplicados en una forma más generalizada, llegando al concepto de “Software as a Service” (Software como Servicio) y la tendencia hacia la implementación de una Arquitectura Orientada a Servicios o SOA (por sus siglas en inglés). La fuerza impulsora detrás de SOA es la realización de que para permanecer competitivo y rentable, las soluciones de negocio de una organización deben ser capaces de adaptarse rápidamente a los cambios de ambiente del negocio. Los principios de arquitectura claves detrás de SOA son la habilidad para reusar activos existentes de software cuando sea posible y exponer la funcionalidad de estos activos como un conjunto de servicios.

Un servicio provee un conjunto de operaciones bien definido que soporta la lógica de negocio de la organización. Un desarrollador puede implementar estas

⁶ Sharp, J. (2010). *Windows Communication Foundation 4 Step by Step*. Microsoft.

operaciones invocando nuevos y existentes activos de software, y componiendo estos elementos en cualquier forma que sea necesaria para satisfacer los requerimientos de la organización. Un servicio oculta los detalles de su implementación, y los creadores del servicio pueden componer nuevas funcionalidades al combinar de forma transparente llamadas a otras aplicaciones o servicios. Las claves para implementar un servicio de forma exitosa son:

- Proveer una interfaz reutilizable y extensible basada en estándares bien definidos para maximizar la interoperabilidad.
- Proveer un ambiente de hospedaje escalable dentro del cual un servicio pueda responder rápidamente a las peticiones del usuario, incluso bajo condiciones de carga extrema.

Aquí es cuando el sistema operativo de Microsoft Windows y WCF hacen lo suyo.

WCF provee un modelo con el cual se pueden implementar servicios que se adaptan a estándares y estilos comúnmente aceptados, incluyendo SOAP, XML, y JSON. Adicionalmente, WCF soporta varias de las tecnologías específicas de Microsoft para construir componentes, tal como Enterprise Services y Microsoft Message Queue (MSMQ), y soporta un modelo de programación unificado para muchas de estas tecnologías. Esto permite la construcción de soluciones que son tan independientes como es posible desde el mecanismo subyacente usado para conectar los servicios y hacerlos accesibles a aplicaciones y servicios implementados con tecnologías ajenas a Microsoft.

Es difícil, si es que no imposible, separar completamente la estructura de programación de una aplicación o servicio de su infraestructura de comunicación, pero WCF permite acercarse mucho al alcance de este objetivo la mayor parte del tiempo. Adicionalmente, al usar WCF, se puede mantener la compatibilidad hacia atrás con muchas de las tecnologías precedentes. Por ejemplo, una aplicación cliente WCF se puede comunicar fácilmente con un servicio Web creado usando WSE.

El sistema operativo Windows puede implementar de una plataforma escalable, segura, y robusta para una SOA. Windows Server 2008 está altamente optimizado para este ambiente, y se pueden crear clusters extensibles de computadores Windows Server haciendo uso de hardware básico de bajo costo.

2.2. DEFINICIÓN DE ESTRUCTURA DE FINANCIER Y CONFIGURACIONES GENERALES CON TECNOLOGÍA WCF

Con el fin de definir los cambios a realizar en las soluciones originales de servidor, proxy y cliente, se comenzó por seleccionar las principales configuraciones de WCF a partir de las necesidades específicas de Contactor.

Entre los principales elementos de WCF que se tuvieron en cuenta para realizar la actualización de tecnología se encuentran:

- **Contratos de servicio:** Se separaron los contratos de servicio de su implementación mediante la definición de las interfaces en un proyecto separado llamado WCFInterfaces, el cual contiene una carpeta por cada módulo existente en Financial, y dentro de cada uno de ellos se ubican las interfaces correspondientes a los servicios que compone cada módulo de Financial.
- **Transporte y Binding:** Teniendo en cuenta que la aplicación se encuentra disponible solamente para ser accedida desde la Intranet corporativa, que todos los equipos de cómputo de los colaboradores usan sistemas operativos Windows y que no se soportarán llamadas en modo desconectado se eligió TCP como el mecanismo de transporte a utilizar.

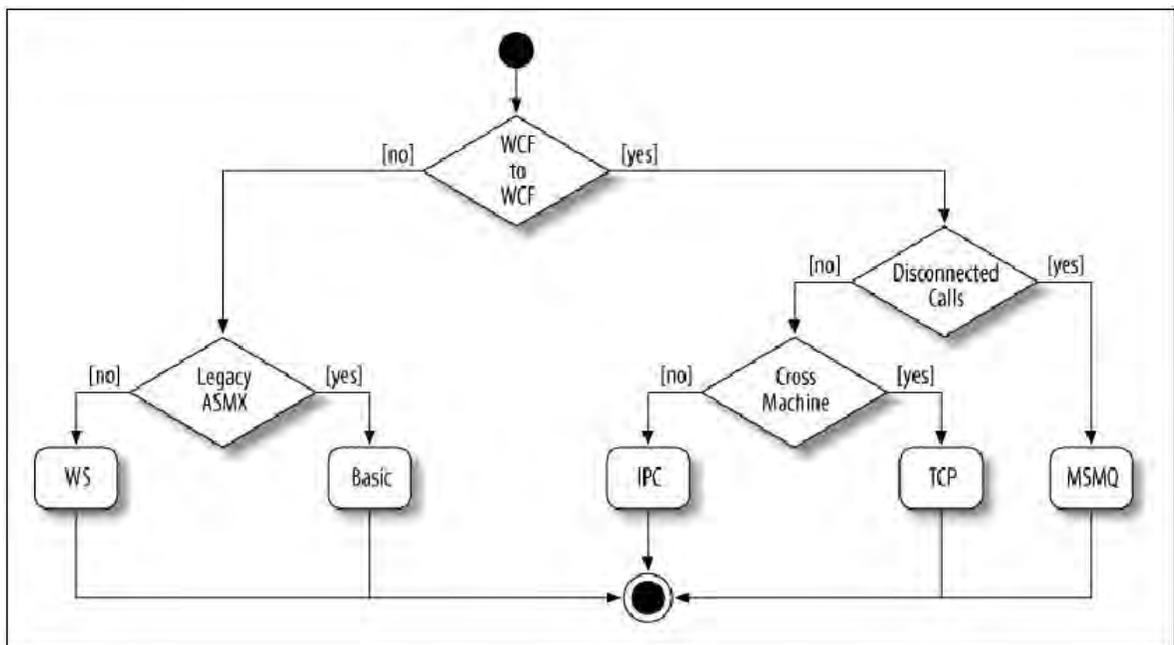


Figura 3 Elección de transporte

Con la elección de TCP como el mecanismo de transporte a utilizar, y los datos de referencia de la figura 3 se define que NetTcpBinding debe ser el binding a implementar en Financial Business System.

Name	Transport	Encoding	Interoperable
BasicHttpBinding	HTTP/HTTPS	Text, MTOM	Yes
NetTcpBinding	TCP	Binary	No
	IPC	Binary	No
NetNamedPipeBinding			
WSHttpBinding	HTTP/HTTPS	Text, MTOM	Yes
NetMsmqBinding	MSMQ	Binary	No

Figura 4 Transportes y codificaciones para bindings comunes

- Hospedaje:** Teniendo en cuenta que el servidor de Financial cuenta con IIS 7.5 y de acuerdo con las recomendaciones especificadas en la figura 4, se optó por alojar la aplicación haciendo uso de IIS(Internet Information Services) y WAS (Windows Activation Service). WAS es un servicio del sistema que se encuentra disponible en Windows Vista, Windows Server 2008, y Windows 7 (o posteriores). WAS es un motor de hospedaje de propósito general, el cual puede hospedar sitios web, pero de igual forma puede hospedar otros servicios, permitiendo usar cualquier mecanismo de transporte, tal como TCP, IPC o MSMQ. Debido a que WAS es un servicio del sistema, no se necesita lanzar previamente el proceso de hospedaje de los servicios; cuando la primera llamada del cliente llegue, WAS la interceptará, lanzará un proceso para albergar el servicio, y le dirigirá la llamada.

WAS ofrece muchas ventajas sobre self-hosting, incluyendo application pooling, reciclaje, gestión de tiempos de inactividad, gestión de la identidad, y aislamiento.

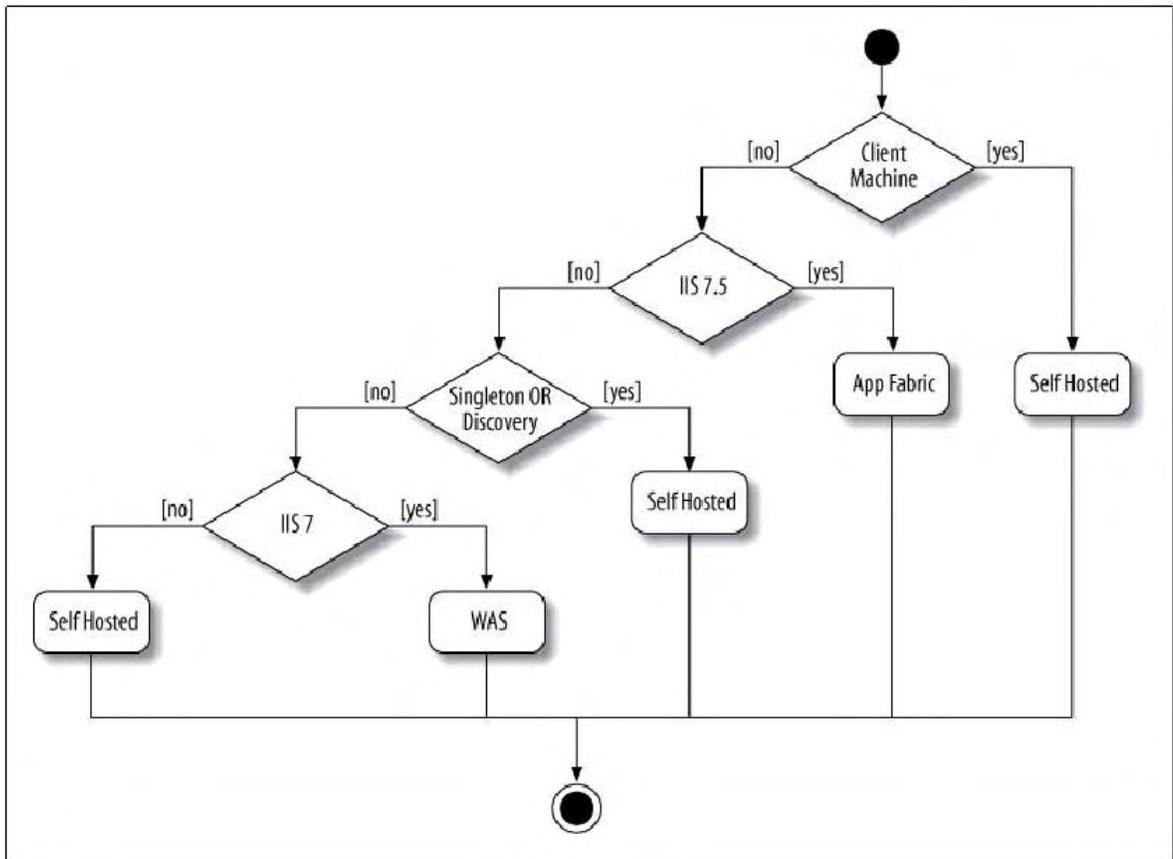


Figura 5 Elección de hosting

- **Seguridad:** WCF provee varios mecanismos de seguridad para que la transmisión de datos entre el cliente y el servidor se realice de forma confiable. Para el caso específico de Financial Business Systems se han implementado los siguientes mecanismos:
 - Verificación de identidad: a través del uso de certificados X509 tanto la aplicación cliente y como la aplicación servidor pueden verificar la identidad de la otra parte. También es conocida como Two Way authentication.
 - Encriptación de mensajes: permite mantener la confidencialidad en la comunicación entre la aplicación cliente y el servicio.
 - Message Hashing: previene la corrupción del mensaje durante su tránsito entre cliente y servidor.
 - Reliable Messaging: asegurar la entrega verificable de mensajes.

Aprovechando la implementación de seguridad relativamente poco intrusiva de WCF, se pueden separar los aspectos relacionados con seguridad de la lógica de la aplicación, permitiendo modificar o extender la seguridad del sistema sin necesidad de reescribir grandes fragmentos de código. Por lo tanto, en caso de que surja la necesidad de implementar nuevas técnicas de seguridad, el cambio no afectará la lógica del negocio ni provocará una pausa significativa de los servicios de Financial Business System.

Una vez seleccionados los elementos de configuración, se determinaron los cambios a implementar en las diferentes soluciones que componen Financial.

Inicialmente se trabajó sobre la solución FBSServidor, la cual ahora con la tecnología WCF tiene la estructura que se muestra en las figuras 6 y 7.

Los proyectos creados y/o modificados son:

- WCFInterfaces: contiene la definición de las interfaces que se implementan en los servicios web.
- FBSServicios: implementación de los servicios web que son consumidos por los clientes de Financial.
- FBSMensajería: clases que definen la estructura de los mensajes intercambiados en los servicios web.
- PagosATH: proyecto que permite registrar en Financial el pago de cuotas desde ATH.
- SiesaIntegracion: proyecto que permite migrar la información contable desde Financial a Siesa, sincroniza los datos de terceros y la emisión de cheques.
- FBSExcel: proyecto que permite incluir en Excel funcionalidades extras relacionadas con crédito, cartera y contabilidad a partir de datos y funcionalidades existentes en Financial.
- FBSWcfServices: sitio web que contiene los datos de los servicios web que expone Financial a sus clientes.

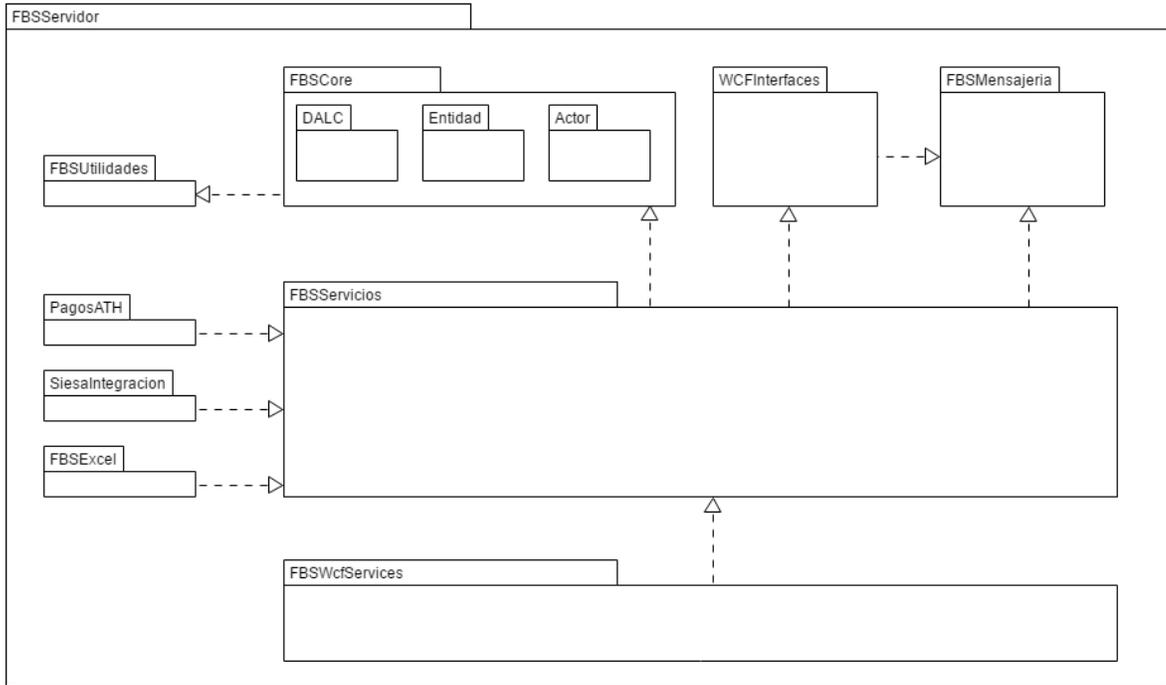


Figura 6 Estructura solución FBSServidor

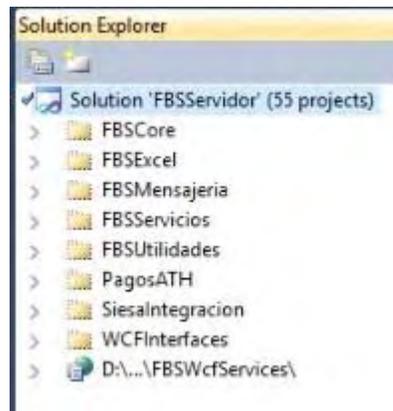


Figura 7 Composición de solución FBSServidor

Una vez creado el web site con los servicios web de Financial, se actualizó la referencia a los servicios en la solución FBSProxies.

En esta solución de Visual Studio existen los proyectos que contienen la referencia a los servicios web de Financial con sus correspondientes metadatos.

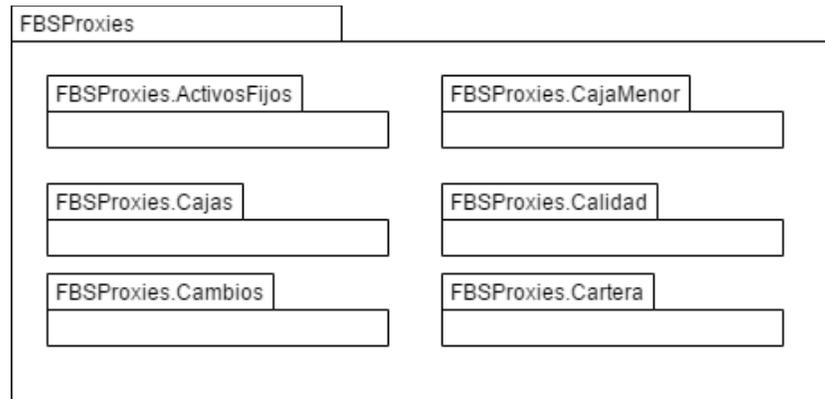


Figura 8 Estructura solución FBSProxies



Figura 9 Composición solución FBSProxies

Con la estructura de la solución FBSProxies definida, se procedió a reestructurar la solución FBSCiente, sobre la cual se trabajó principalmente en el proyecto FBSProxies, esto debido a que las clases de este proyecto se utilizan como intermediarias en las llamadas a los servicios web tanto síncronas como asíncronas, por lo tanto, su adecuada encapsulación permitió disminuir el código a implementar en la capa de presentación de la solución.

Las principales modificaciones realizadas fueron:

- FBSProxies: se redefinió la instanciación de las clases intermediarias en las llamadas a los servicios web.
- Se modificaron las llamadas asíncronas a los servicios web debido a la actualización de las clases utilizadas en este proceso.
- Se implementó el manejo de sesiones independientes para la interacción con cada uno de los servicios web.
- Se redefinió el manejo de excepciones en toda la solución debido a que ahora se utiliza la clase FaultException para validar las llamadas a los servicios web.

La estructura de la solución FBSCiente actualizada a WCF se presenta a continuación:

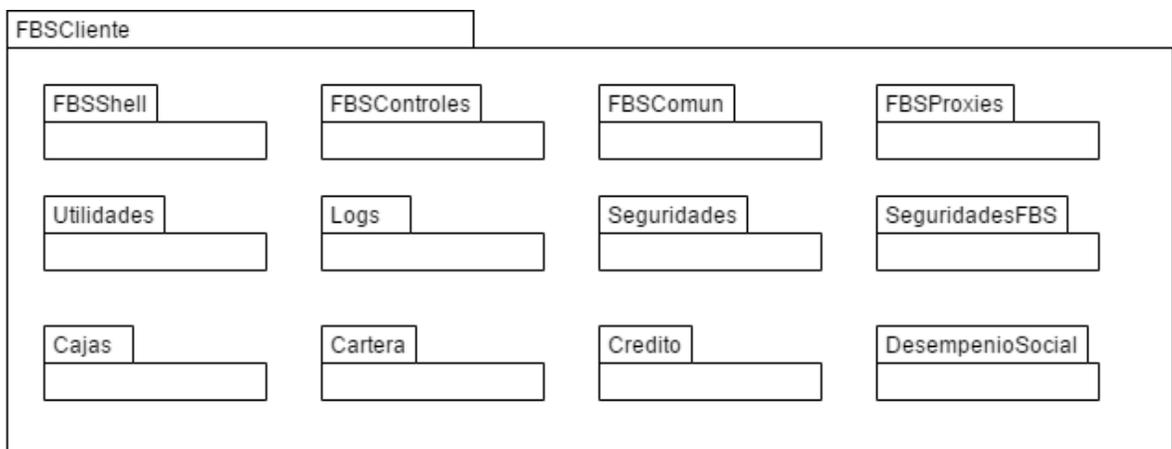


Figura 10 Estructura solución FBSCiente



Figura 11 Composición solución FBSCliente

2.3. AUTOMATIZACIÓN DE PROCEDIMIENTOS

Durante el proceso de migración de código de las diferentes soluciones que componen Financiamiento fue necesario hacer uso de algunas herramientas que permitieran automatizar tareas que se debían ejecutar sobre la mayoría de los archivos existentes, por tal motivo, después de realizar la respectiva búsqueda, se eligió usar el programa Rename-It! para renombrar archivos en lote y por otra parte, se decidió utilizar Macros de Visual Studio para acceder a secciones específicas del código para realizar modificaciones de forma automática.

2.3.1. Rename-It! es un software usado en sistemas operativos Windows que permite renombrar automáticamente y de forma segura miles de archivos y carpetas a través del uso de expresiones regulares y muchos otros tipos de filtros.

Algunas de las principales características de Rename-It!, son:

- Permite renombrar archivos desde el Explorador de Windows.
- Acepta el uso de expresiones regulares y comodines.
- Provee un sistema de enumeración avanzado (01, 02, 03,...)
- Soporta codificación Unicode.
- Acepta el uso de etiquetas MP3, ID3, v1 y v2.
- Permite el uso simple y avanzado con uso de filtros predefinidos.
- Revisa el nombre de archivos antes de ejecutar el renombrado y permite realizar ajustes manuales.

El proceso de renombrado es realizado con el uso de alguno de los siguientes filtros:

- Buscar y reemplazar: reemplaza una cadena por otra. Soporta expresiones regulares y comodines.
- Mayúsculas y minúsculas: permite cambiar el nombre del archivo a mayúsculas o minúsculas.
- Cortar: solamente conserva una parte del nombre del archivo.
- Etiqueta ID3: para archivos MP3, lee la información ID3 y renombra el archivo a partir de los criterios definidos por el usuario.

Este software fue utilizado en el proceso de actualización de Financial para renombrar archivos en lote y para modificar las extensiones de algunos de ellos para ajustarlos a los tipos de archivos utilizados por WCF.

2.3.2. Macros de Visual Studio. Un macro es una serie de comandos e instrucciones que se agrupan en forma de un mismo comando para ejecutar una tarea automáticamente. Los macros permiten automatizar acciones repetitivas. Visual Studio incluye un IDE de Macros, el cual es usado para crear, manipular, editar y ejecutar solamente macros. El IDE de Macros se encuentra separado del IDE Visual Studio.

El uso de macros de Visual Studio fue de gran valor para el proceso de actualización de tecnología de Financial debido a que permitió automatizar muchas de las tareas que se debían ejecutar de forma repetitiva durante la migración de código de las diferentes soluciones de Visual Studio.

El IDE de macros, junto a las librerías Env.DTE, Env.DTE80, Env.DTE90, entre otras permitieron recorrer cada uno de los proyectos y archivos que componen las soluciones de Visual Studio, además brindaron la posibilidad de acceder a segmentos de código específicos, tales como la declaración de espacios de nombres (namespaces), de clases, de variables, de métodos con sus correspondientes coordenadas de inicio y fin. Todos estos datos posibilitaron la reestructuración de código en todas las soluciones de Financial de forma ágil y segura, disminuyendo considerablemente el tiempo necesario para completar la actualización de tecnología, y evitando la ocurrencia de errores de digitación.

Algunas de las tareas que permitió automatizar el uso de macros fueron:

- Generación de los archivos del proyecto WCFInterfaces a partir de la definición de los métodos que se encontraban publicados anteriormente con la tecnología WSE.
- Inclusión de las cabeceras de clases y propiedades en las clases de mensajería para el proyecto FBSMensajería.
- Eliminación de las cabeceras de WSE existentes en la declaración de los servicios web.
- Actualización en lote de las referencias a los servicios web en la solución FBSProxies.
- Reestructuración de la definición de las clases de FBSProxies en la solución FBSCliente.

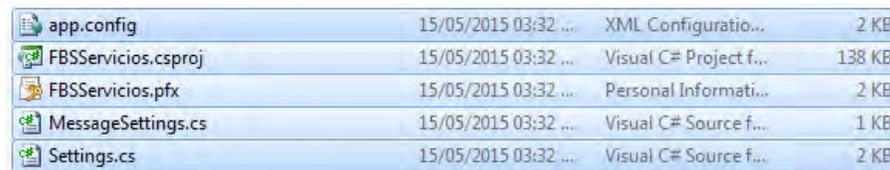
2.4. MIGRACIÓN SOLUCIÓN SERVIDOR

Para realizar la migración a WCF de la solución servidor se desarrolló la guía que se detalla a continuación:

2.4.1. Creación de Interfaces

2.4.1.1. Copiar FBSServicios. Copiar la carpeta FBSServicios (Copiar versión compilada) a una carpeta Temp en la cual se realizara el proceso de renombrar archivos, aumentando la letra I al final del archivo así: *AccesoriosWS.cs* se renombra a *AccesoriosWSI.cs*

- Para las interfaces, el nombre del archivo será AccesoriosWSI pero el nombre de la clase será IAccesoriosWS.
- Verificar que el nombre de las clases sea el mismo que el nombre del archivo.
- Eliminar los elementos innecesarios:
 - Carpeta *Properties*
 - Carpeta *Web References* (DataCredito)
 - Carpeta *obj*
 - Archivo *app.config*
 - Archivo *FBSServicios.csproj*
 - Archivo *FBSServicios.pfx*
 - Archivo *MessageSettings.cs*
 - Archivo *Settings.cs*



app.config	15/05/2015 03:32 ...	XML Configuratio...	2 KB
FBSServicios.csproj	15/05/2015 03:32 ...	Visual C# Project f...	138 KB
FBSServicios.pfx	15/05/2015 03:32 ...	Personal Informati...	2 KB
MessageSettings.cs	15/05/2015 03:32 ...	Visual C# Source f...	1 KB
Settings.cs	15/05/2015 03:32 ...	Visual C# Source f...	2 KB

Figura 12. Elementos innecesarios para renombrar

2.4.1.2. Renombrar archivos en lote. Para renombrar los archivos en lote se utiliza el programa Rename-It; este software para la plataforma Windows permite definir expresiones regulares para renombrar archivos en lote.

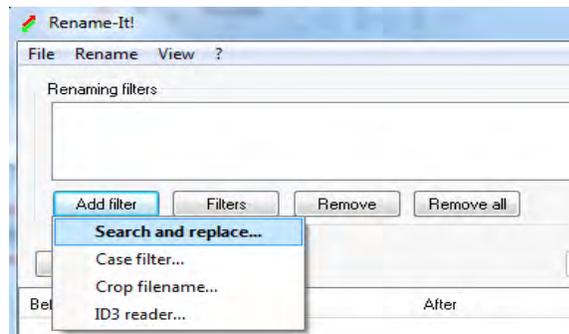


Figura 13 Rename-It - Elección de filtro

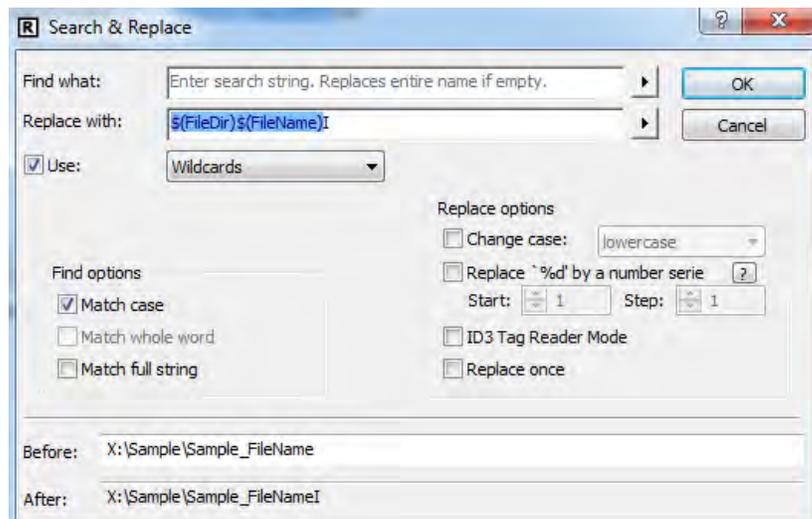


Figura 14 Rename-It - Expresión de reemplazo

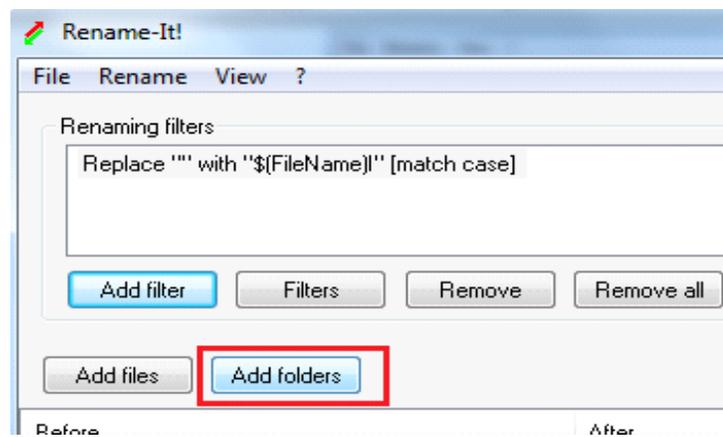


Figura 15 Rename-It - Elección de carpetas

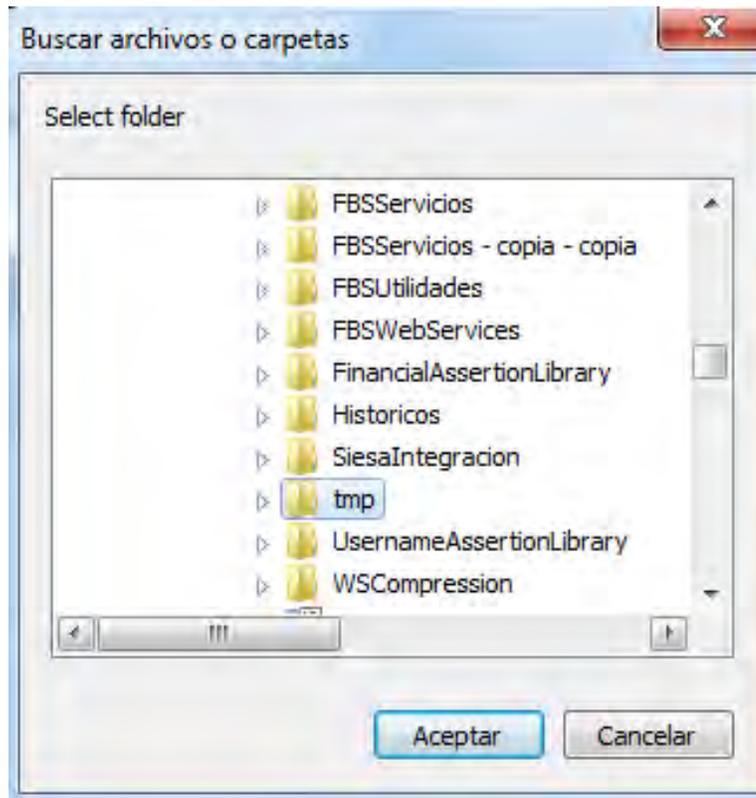


Figura 16 Rename-It - Elección de carpetas 2

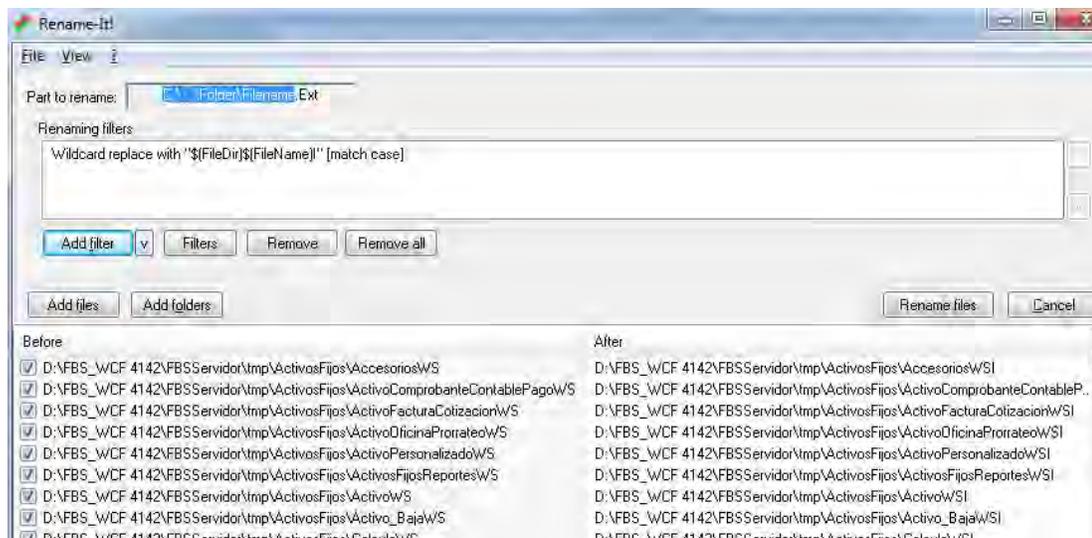


Figura 17 Rename-It - Resumen de configuración

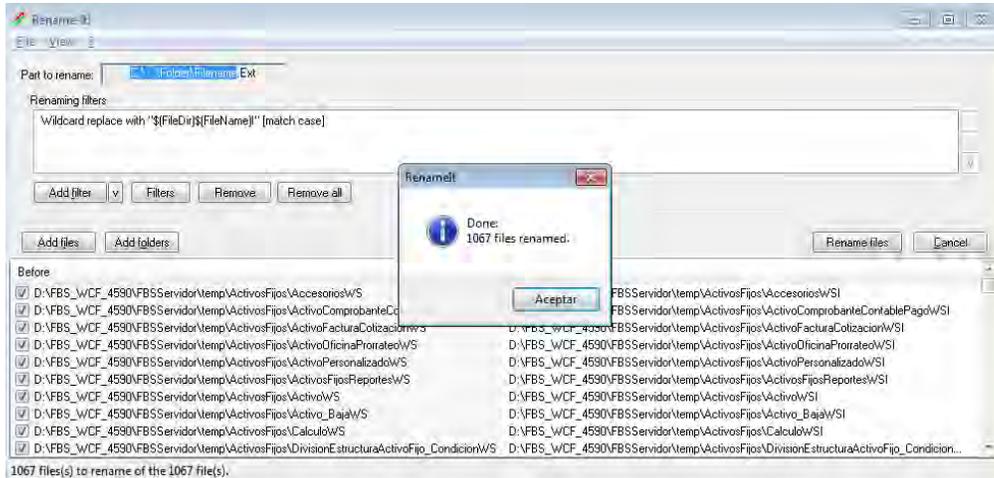


Figura 18 Rename-It - Proceso terminado

Una vez realizado este proceso los nombres de los archivos serán tal como se muestran en la siguiente figura:

Equipo > Informacion (D:) > FBS_WCF_4590 > FBSServidor > temp > ActivosFijos				
Incluir en biblioteca > Compartir con > Grabar > Nueva carpeta				
	Nombre	Fecha de modifica...	Tipo	Tamaño
itios				
cargas	AccesoriosWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	9 KB
itorio	Activo_BajaWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	13 KB
rs recientes	ActivoComprobanteContablePagoWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	26 KB
	ActivoFacturaCotizacionWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	21 KB
tecas	ActivoOficinaProrratoWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	7 KB
umentos	ActivoPersonalizadoWSI.cs	15/05/2015 03:32 ...	Visual C# Source f...	5 KB

Figura 19 Rename-It - Archivos renombrados

2.4.1.3. Cargar interfaces a la solución FBSServidor. Crear la carpeta y el proyecto (Class Library) que contendrá las interfaces: WCFInterfaces.

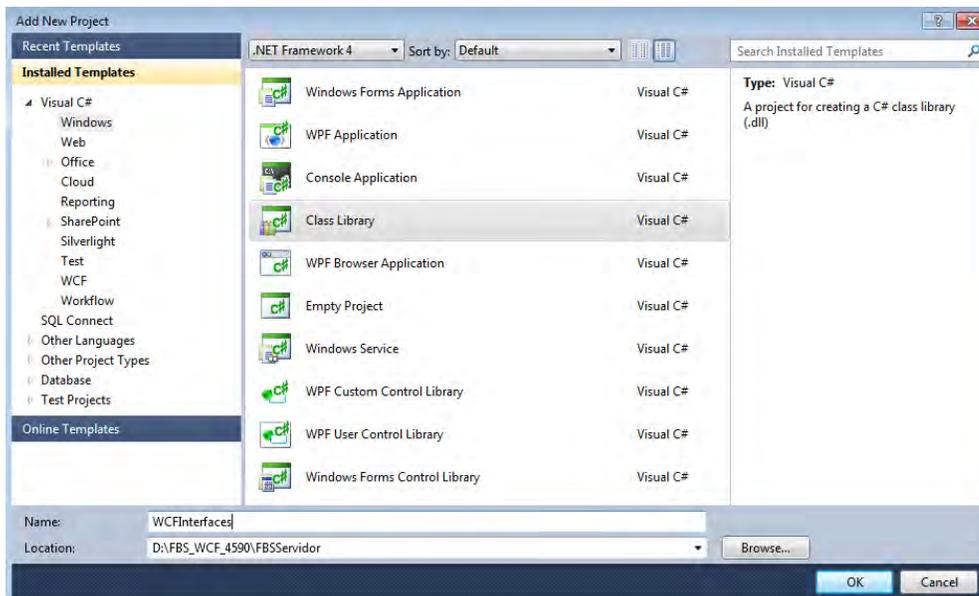


Figura 20 WCFInterfaces - Crear proyecto

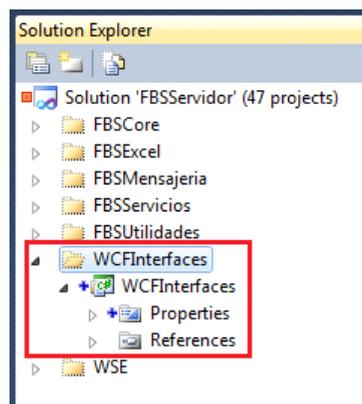


Figura 21 WCFInterfaces - Archivos automáticos a eliminar

- Eliminar la clase que se crea automáticamente al agregar el proyecto WCFInterfaces.
- Copiar las carpetas de Temp/ a la carpeta del proyecto WCFInterfaces.

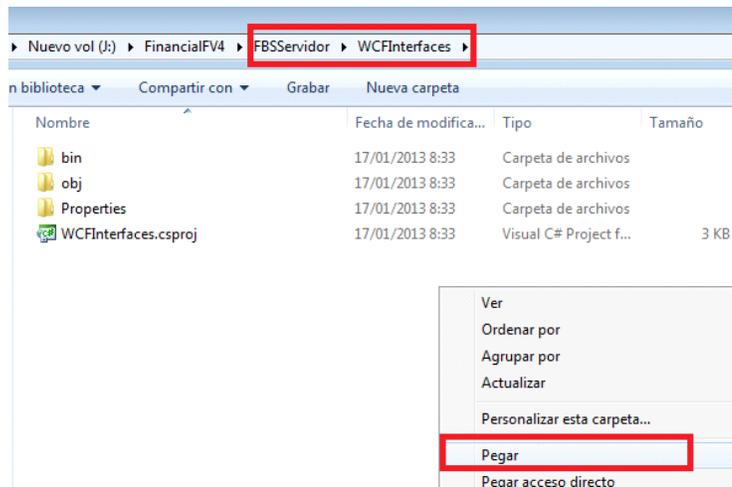


Figura 22 WCFInterfaces - Mover archivos desde temp

- Cargar los archivos en Visual Studio, excepto la carpeta bin y obj.
- Habilitar ver todos los archivos y cargar los archivos excluidos.

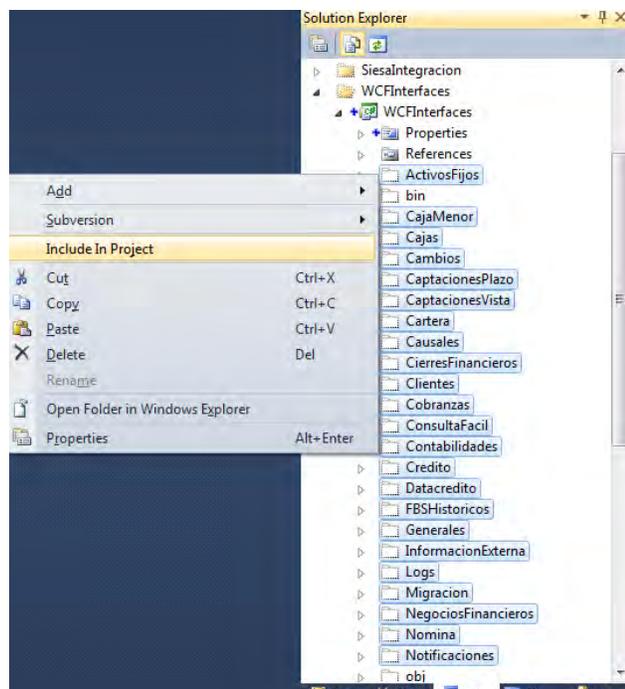


Figura 23 WCFInterfaces - Incluir archivos en proyecto

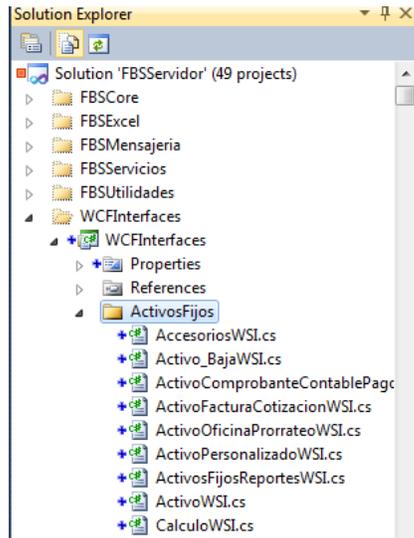


Figura 24 WCFInterfaces - Archivos incluidos

- Después de incluir todos los archivos en el proyecto, se deben excluir aquellos que se encontraban excluidos originalmente. Basarse en el proyecto FBSServicios (habilitando la opción Show all files) para saber cuáles son los archivos que deben permanecer excluidos. El listado de archivos excluidos hasta la revisión 3579 de subversión, se indica en la siguiente tabla:

Tabla 1 WCFInterfaces - Archivos excluidos

MODULO	ARCHIVO
Cartera	TipoPrestamoComponenteCarteraPorcentajeWS.cs
Contabilidades	EstadoFinancieroWS.cs
Datacredito	AlertaWS.cs
Obligaciones	TipoObligacionComponente_ObligacionWS.cs
Personas	PersonaOrganizacionGrupoResultadosProyectoWS.cs

- Guardar todos los cambios realizados hasta el momento.

2.4.1.4. Codificar interfaces. Modificar el contenido de los archivos renombrados para generar la estructura adecuada de las interfaces mediante el uso del Macro del proyecto V4_WCFInterfaces.

- Abrir el archivo ModuloRecorrerProyecto.
- La constante MODO_DEBUG debe tener asignado el valor False.

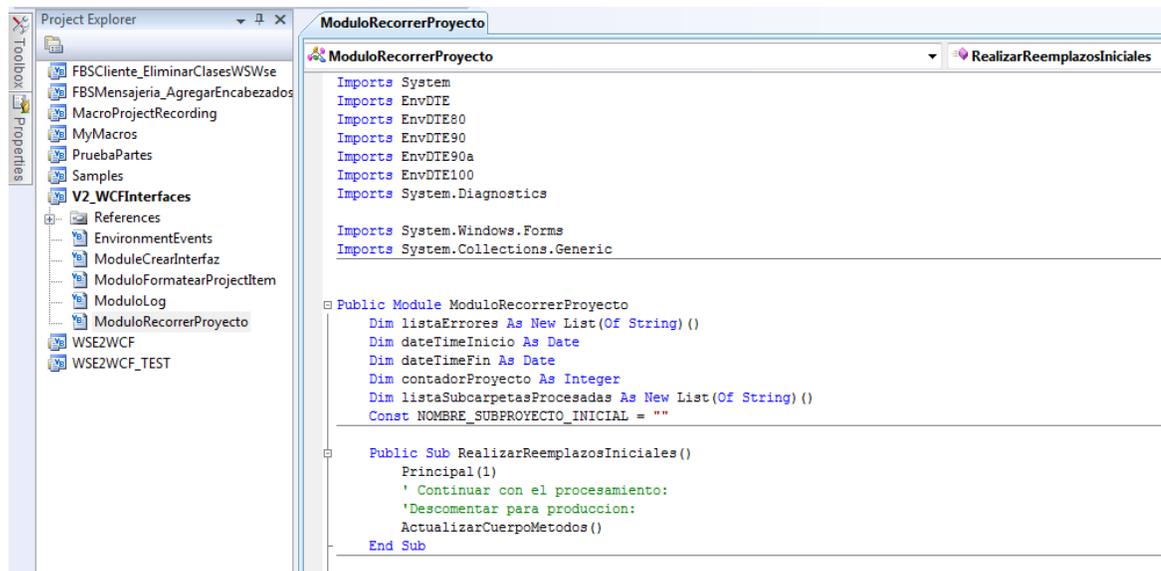


Figura 25 Macro V4_WCFInterfaces - ModuloRecorrerProyecto

- Ubicar el cursor en el inicio del método RealizarProcesamientoCompleto y ejecutar el macro (F5).

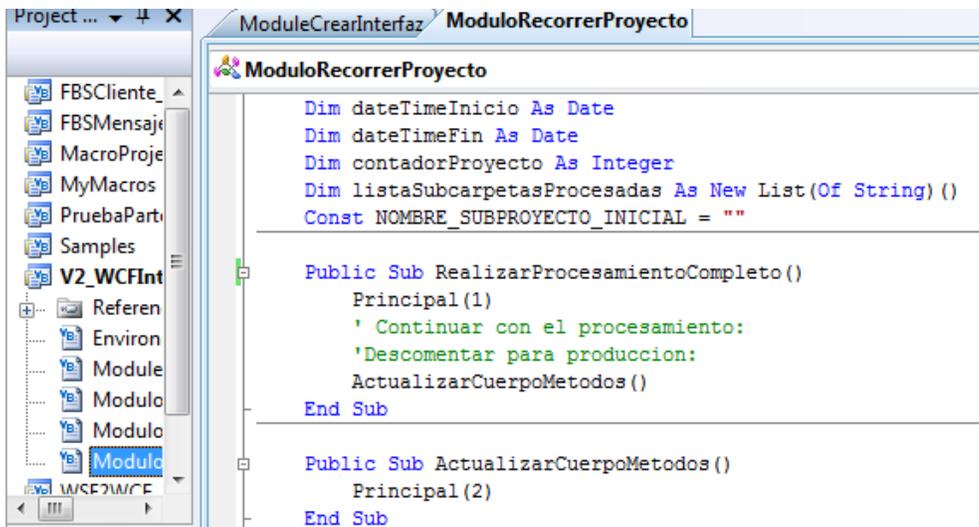


Figura 26 V4_WCFInterfaces - RealizarProcesamientoCompleto

- Al finalizar el proceso (tiempo aproximado 30 minutos) se visualizará una ventana como la que se indica en la siguiente figura.

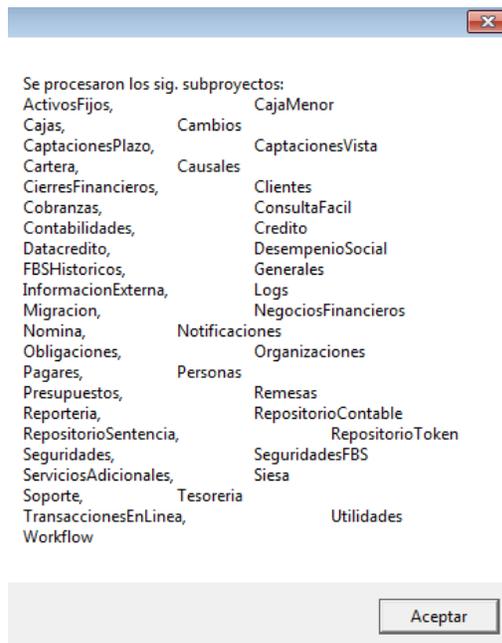


Figura 27 V4_WCFInterfaces - Subproyectos procesados

- Agregar a este proyecto los archivos **MessageSettings.cs** y **RaiseException.cs**. Ambos archivos se encuentran en la carpeta Recursos/WCFInterfaces.

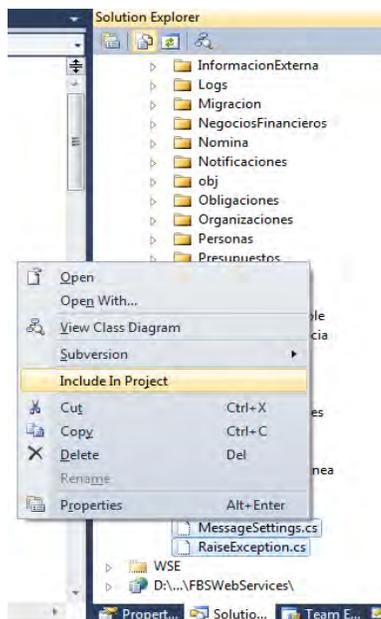


Figura 28 Archivos MessageSettings.cs y RaiseException.cs

2.4.1.5. Agregar referencias. Agregar referencia a los siguientes proyectos/librerías:

- FBSMensajeria
- FBSUtilidades
- FBSCore
- SiesaIntegracion
- System.Transactions
- System.ServiceModel
- System.Runtime.Serialization
- WCFExtras

NOTA: WCFExtras se encuentra incluido en la carpeta Lib del proyecto.

Su página oficial es: <http://wcfextras.codeplex.com/>

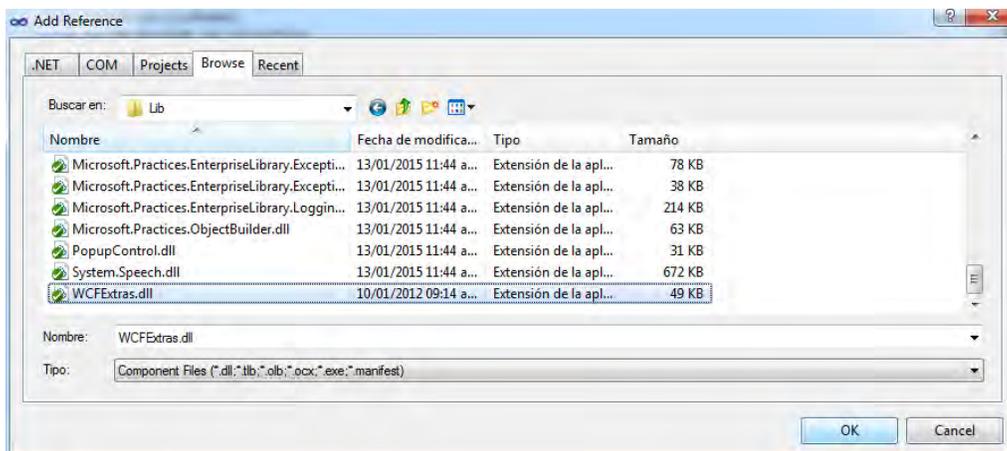


Figura 29 WCFInterfacse - Agregar DLL de WCFExtras

2.4.1.6. Actualizar usings y namespace del proyecto. Buscar y reemplazar por vacío usando los siguientes valores de búsqueda:

- using System.Web.Services;
- using System.Web.Services.Protocols;
- using Microsoft.Web.Services3;

- Actualizar el namespace de los ítems del proyecto.

Buscar: ^namespace FBSServicios

Reemplazar por: namespace WCFInterfaces

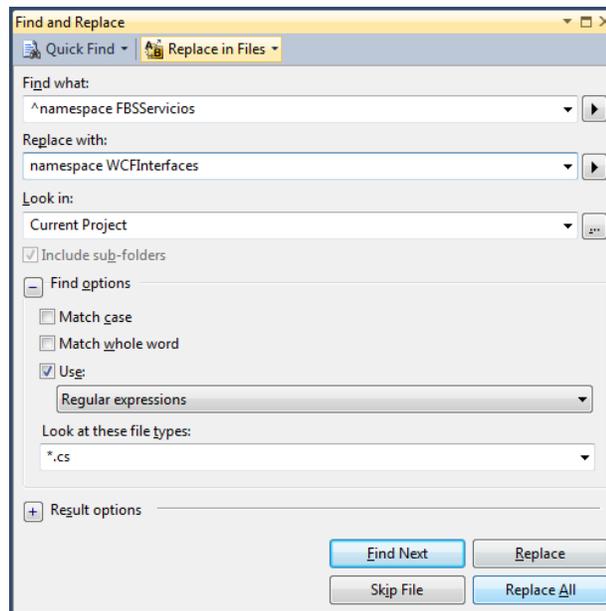


Figura 30 WCInterfaces - Actualizar nombre de namespace

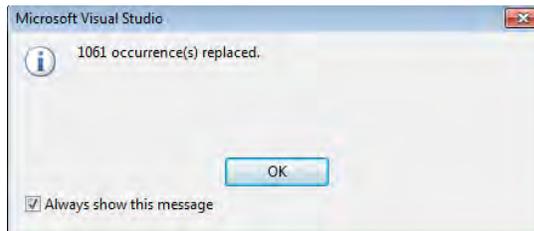


Figura 31 WCInterfaces - Namespace modificado

- Quitar los using de FBSServicios
- Buscar: [^using FBSServicios\.{(.*)}]
- Reemplazar por: [//using FBSServicios.\1]

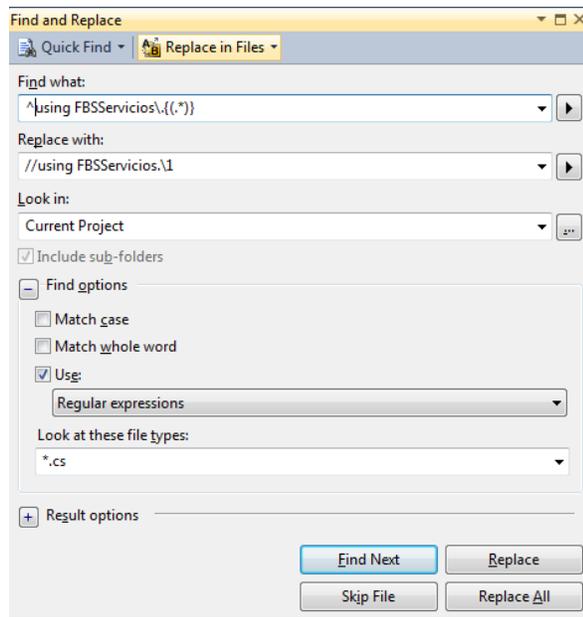


Figura 32 WCFInterfaces - Eliminar importación de FBSServicios

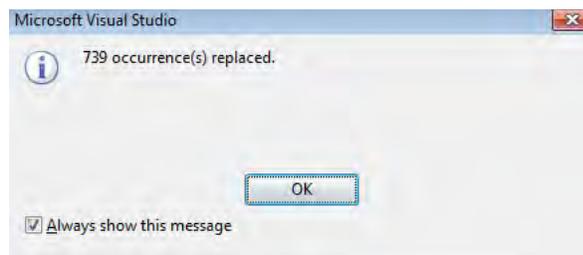


Figura 33 WCFInterfaces - Importación de FBSServicios eliminada

2.4.1.7. Ajustes a archivos específicos. En el archivo SeguridadesFBS/ILoginWS eliminar las cabeceras SoapHeader y FaultContract del método Autenticar.

```

WCFInterfaces.SeguridadesFBS.ILoginWS
{
    /// <summary>
    /// Constructor por defecto
    /// </summary>

    /// <summary>
    /// Autentica un usuario y devuelve recursos para el cliente
    /// </summary>
    /// <param name="mensajeEntrada">Mensaje de entrada con parametros para autentificar al usuario</param>
    /// <returns>Mensaje de salida con los recursos para el cliente de acuerdo al usuario</returns>
    // [SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
    // [FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
    [OperationContract(Name = "Autenticar")]
    AutenticarMS Autenticar(AutenticarME mensajeEntrada);
}
    
```

Figura 34 WCFInterfaces - Eliminar cabeceras de SeguridadesFBS/ILoginWS

- Eliminar declaración de objeto valoresACobrar en Cartera/PrestamoAfectacionWSI.cs

```

PrestamoAfectacionWSI.cs
public object valoresACobrar { get; set; }
}
    
```

Figura 35 WCFInterfaces - Eliminar declaración de objeto valoresACobrar

2.4.1.8 Compilar proyecto WCFInterfaces y corregir errores. Al realizar la compilación del proyecto se pueden presentar errores debido a que en algunos archivos no se reconoce el tipo SoapHeaders.

11 Errors 0 Warnings 0 Messages		
	Description	File
2	The type or namespace name 'SoapHeaders' could not be found (are you missing a using directive or an assembly reference?)	ModalidadPagoWSI.cs
3	The type or namespace name 'SoapHeadersAttribute' could not be found (are you missing a using directive or an assembly reference?)	ModalidadPagoWSI.cs

Figura 36 WCFInterfaces - Posible error por no importación de SoapHeaders

En este caso se debe agregar manualmente el siguiente código:

```
using System;  
using System.ServiceModel;  
using System.Runtime.Serialization;  
using WCFExtras.Soap;  
using System.Net.Security;
```

2.4.2. Modificar proyecto FBSServicios

2.4.2.1. Formatear proyecto. Se deben realizar las siguientes actividades:

- Abrir proyecto de macros: **FormatearArchivosProyecto**
- Abrir archivo *ModuloObtenerItemsProyecto*.
- Verificar la constante `NOMBRE_PROYECTO_FORMATEAR` tenga como valor el proyecto **FBSServicios**.

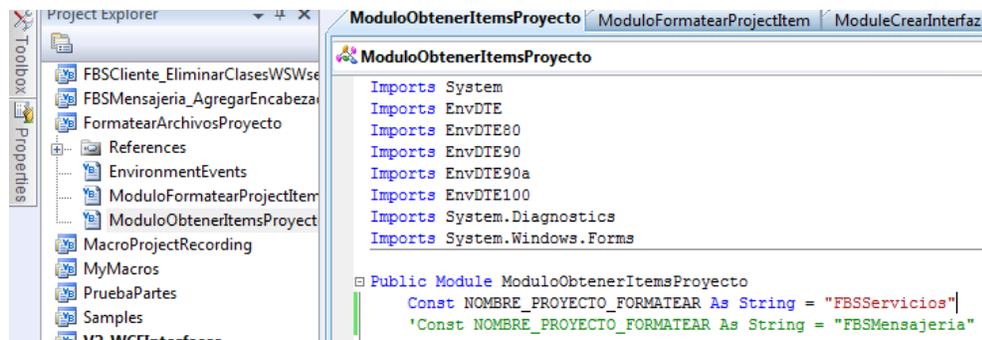


Figura 37 FBSServicios - Macro FormatearArchivosProyecto

Ubicarse en el método principal `ProcesarProyecto()` y ejecutar (F5)

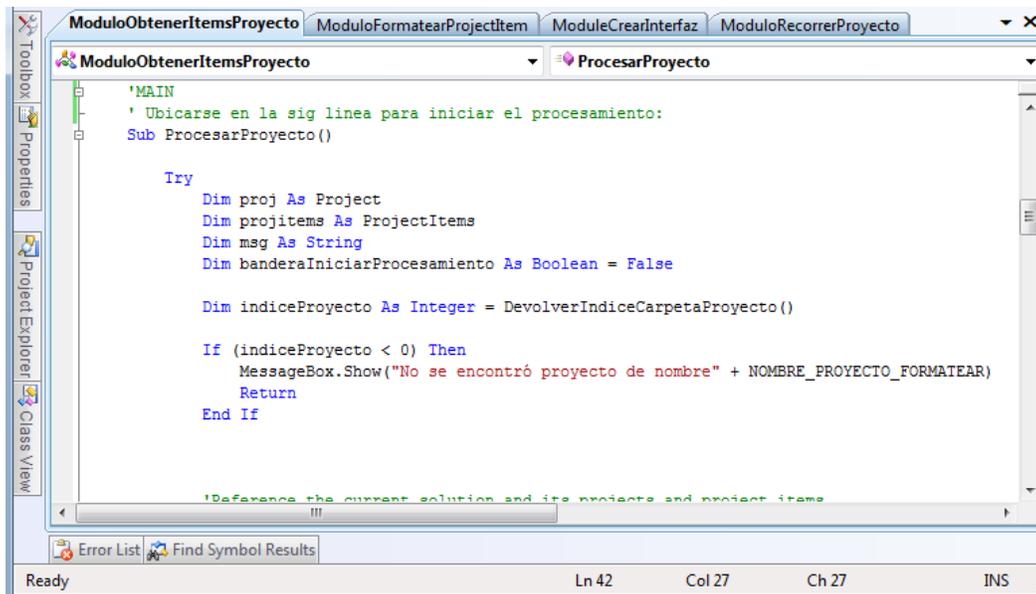


Figura 38 FBSServicios - FormatearArchivosProyecto - ProcesarProyecto

2.4.2.2. Eliminar cabeceras de servicios. Eliminar las cabeceras de las clases y de los métodos de las clases de servicios web del proyecto FBSServicios en la solución original FBSServidor.

- Abrir el macro V4_FBSServicios, luego en el archivo ModuloPrincipal modificar la constante MODO_DEBUG = False, ubicarse en el método Main() y ejecutar (F5).

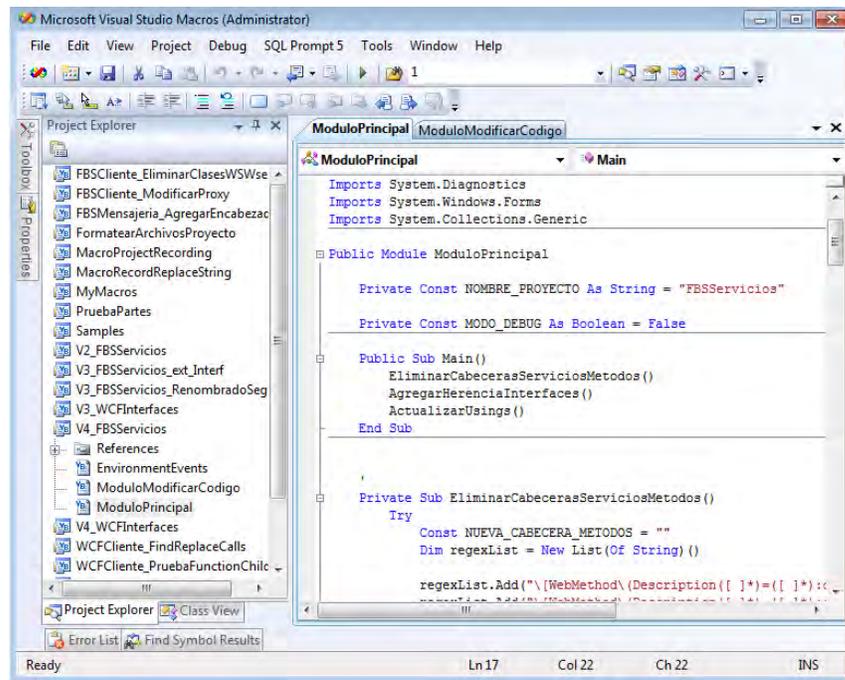


Figura 39 FBSServicios - Ejecutar macro V4_FBSServicios

Tiempo aproximado de ejecución: 20 min.

2.4.2.3. Agregar nuevas clases. Copiar e incluir en el proyecto las clases AuthorizationPolicy.cs, CustomPrincipal.cs e InspectorMensajes.cs, las cuales se encuentran en la ruta Recursos/FBSServicios.

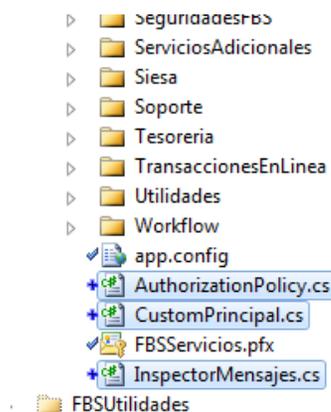


Figura 40 FBSServicios - Incluir clases desde recursos

- Buscar: ^namespace

Reemplazar por: using System.ServiceModel;\n\nnamespace

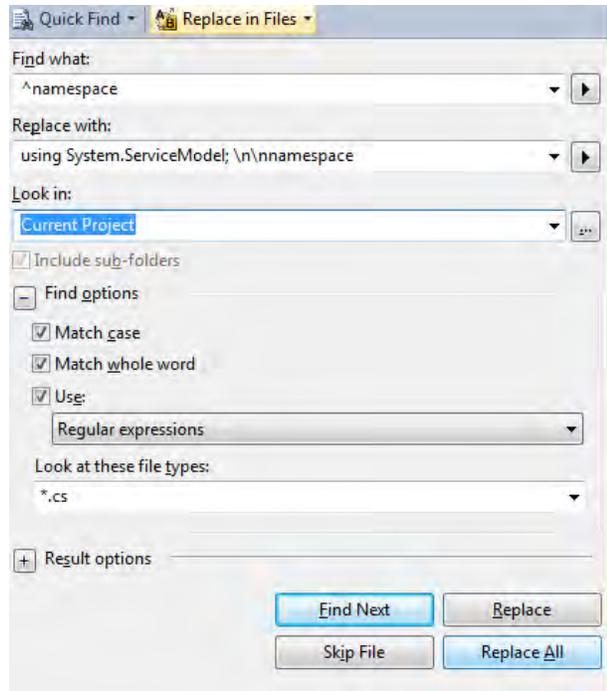


Figura 41 FBSServicios - Agregar using de System.ServiceModel

2.4.2.4. Agregar referencias. En el proyecto FBSServicios agregar la referencia a:

- Proyecto WCFInterfaces.
- Librería System.ServiceModel
- Librería System.IdentityModel
- Librería System.Runtime.Serialization
- WCFExtras (La DLL se encuentra en la carpeta Lib del proyecto)

2.4.2.5. Agregar using a WCFInterfaces. Buscar y reemplazar:

Buscar: [^namespace FBSServicios\.{(.*)}]

Reemplazar por: [using WCFInterfaces.\1;\n\nnamespace FBSServicios.\1]

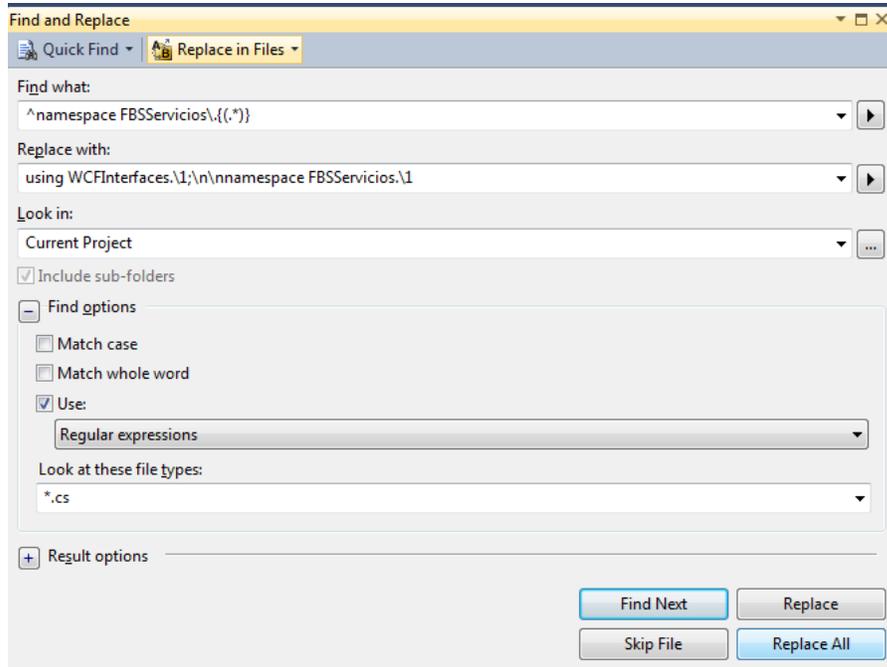


Figura 42 FBSServicios - Agregar importación de WCFInterfaces

2.4.2.6 **Revert a archivos específicos.** Realizar revert a los archivos:

- FBSServicios/Properties/Settings.settings/Settings.Designer.cs.
- FBSServicios/Credito/Database.
- FBSServicios/Web References/ServicioHistoriaCredito.

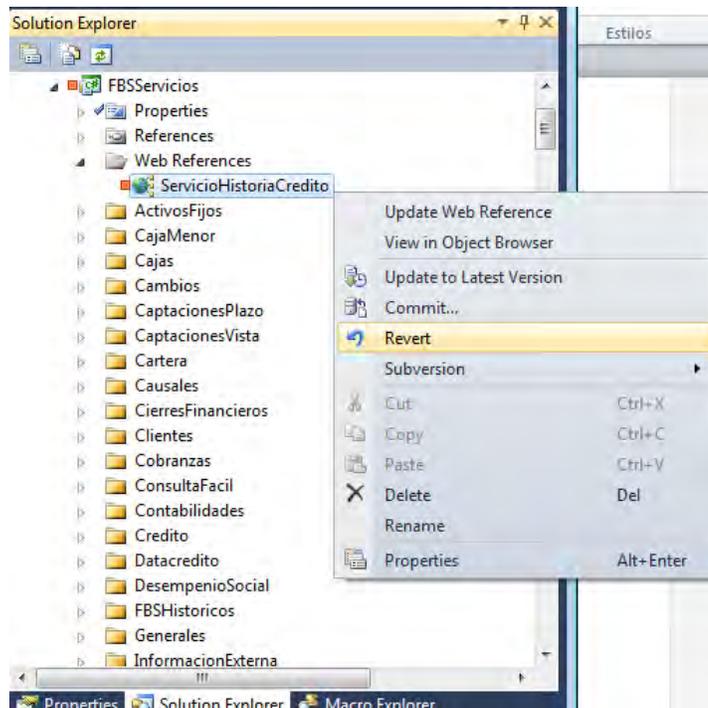


Figura 43 FBSServicios - Revert a referencia web de data crédito

2.4.2.7. Correcciones manuales. Agregar el código #endregion en FBSServicios/Personas/RegimenSaludWS.cs al terminar la etiqueta summary tal como se indica en la siguiente figura.

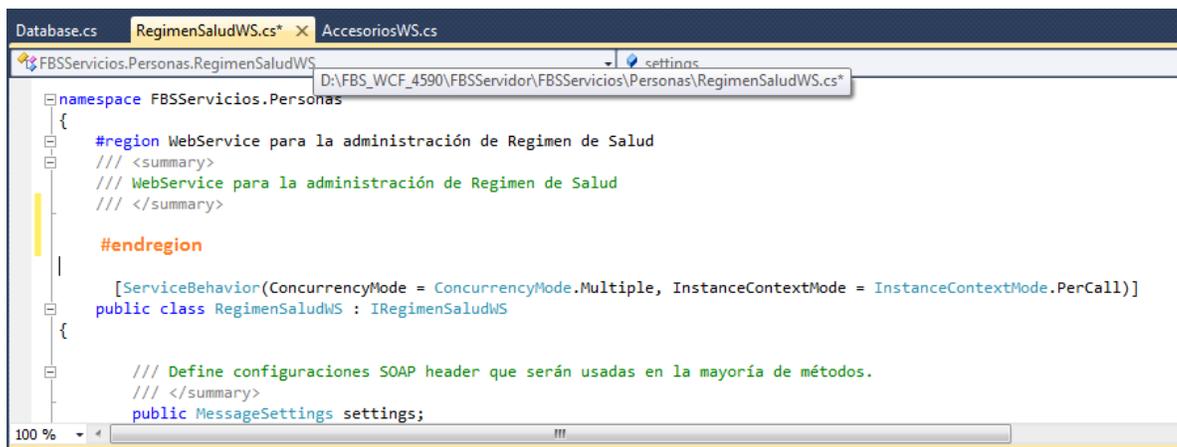
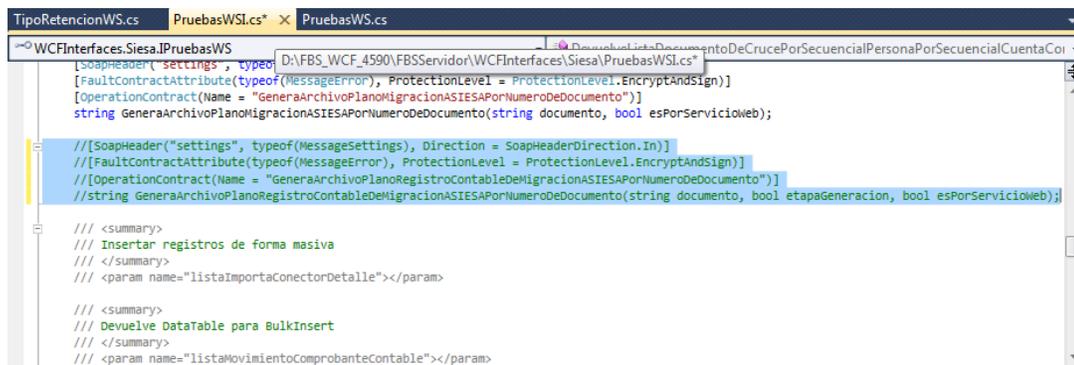


Figura 44 FBSServicios - Agregar #endregion a Personas.RegimenSaludWS.cs

2.4.2.8. Quitar métodos de la interfaz. Comentar los métodos:

- ProcesaFondeoAutomatico del archivo
WCFInterfaces/Credito/SolicitudMaestroWSI.cs
- AfectacionConCheque del archivo
WCFInterfaces/Nomina/EmpleadoVacacionSolicitudWSI.cs
- AfectacionConOrdenPago del archivo
WCFInterfaces/Nomina/EmpleadoVacacionSolicitudWSI.cs
- ProcesaGuardar del archivo
WCFInterfaces/Obligaciones/ObligacionComponenteObligacionWSI.cs
- GeneraArchivoPlanoRegistroContableDeMigracionASIESAPorNumeroDeDocumento del archivo WCFInterfaces/Siesa/PruebaWSI.cs

Para el último caso, la interfaz quedaría así:



```
WCFInterfaces.Siesa.IPruebasWS
[SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
[FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
[OperationContract(Name = "GeneraArchivoPlanoMigracionASIESAPorNumeroDeDocumento")]
string GeneraArchivoPlanoMigracionASIESAPorNumeroDeDocumento(string documento, bool esPorServicioWeb);

//[[SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
//[[FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
//[[OperationContract(Name = "GeneraArchivoPlanoRegistroContableDeMigracionASIESAPorNumeroDeDocumento")]
//string GeneraArchivoPlanoRegistroContableDeMigracionASIESAPorNumeroDeDocumento(string documento, bool etapaGeneracion, bool esPorServicioWeb);

/// <summary>
/// Insertar registros de forma masiva
/// </summary>
/// <param name="listaImportaConectorDetalle"></param>

/// <summary>
/// Devuelve DataTable para BulkInsert
/// </summary>
/// <param name="listaMovimientoComprobanteContable"></param>
```

Figura 45 FBSServicios - Ejemplo método que no es servicio web

2.4.2.9. MessageSettings. Agregar la variable de clase MessageSettings en el archivo FBSServicios/Cartera/AuxilioComercialGerenteZonaWS.cs

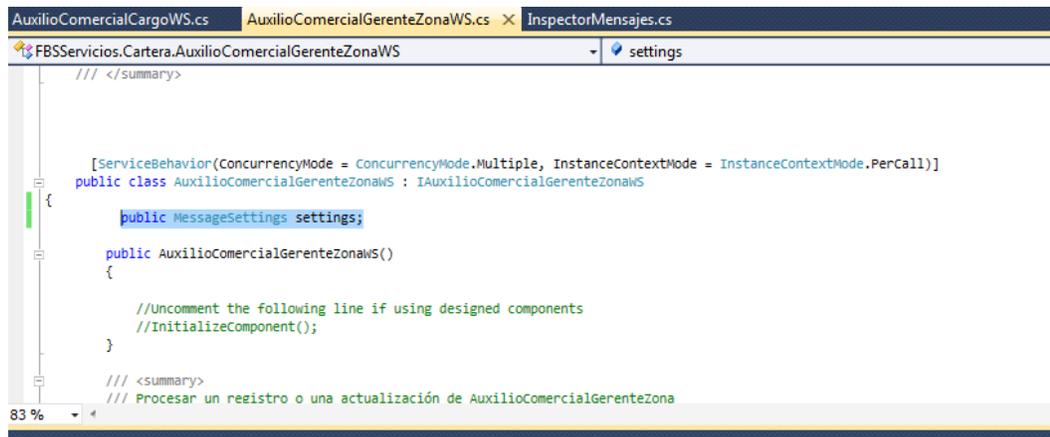


Figura 46 FBSServicios - Agregar variable MessageSettings

- Actualizar la instanciación e inicialización de la variable MessageSettings

Buscar [public MessageSettings settings;]

Reemplazar por [public MessageSettings settings = WCFExtras.Soap.SoapHeaderHelper<MessageSettings>.GetInputHeader("settings");]

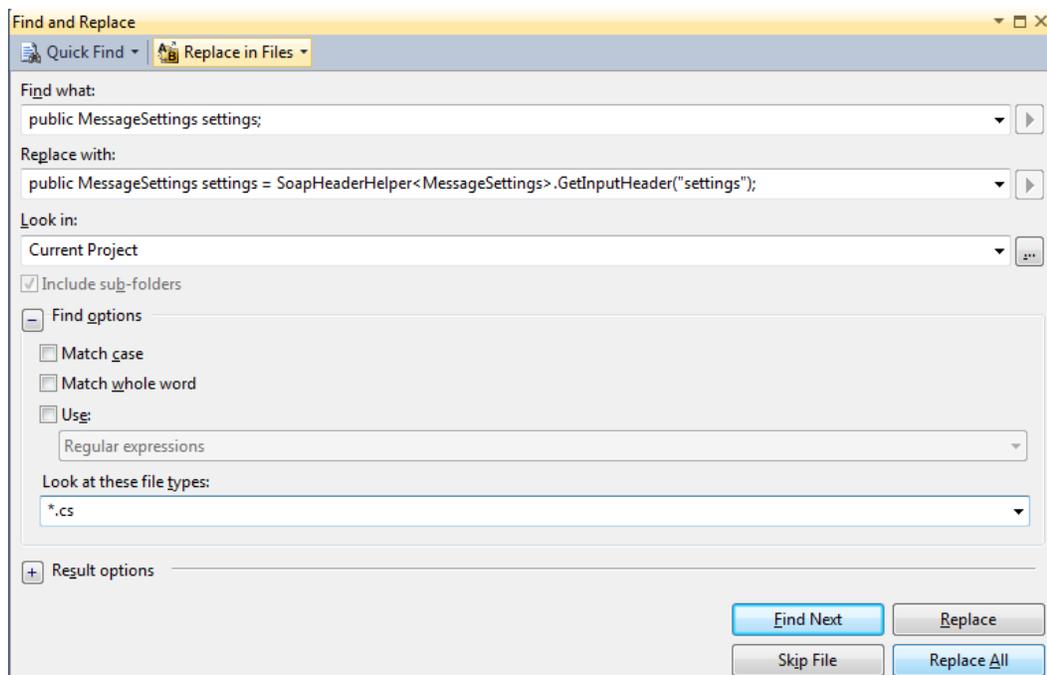


Figura 47 FBSServicios - Actualizar instanciación de variable MessageSettings

- Excluir el archivo MessageSettings de FBSServicios

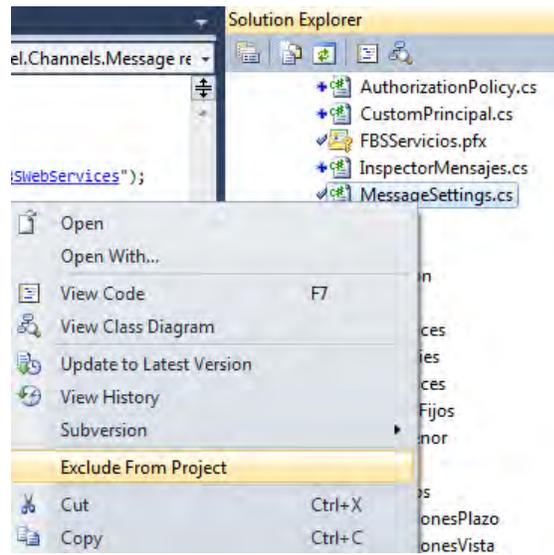


Figura 48 FBSServicios - Excluir archivo MessageSettings

2.4.2.10. Agregar using de WCFInterfaces. En el proyecto FBSServicios:

Buscar: [using WCFInterfaces]

Reemplazar por: [using WCFInterfaces;\nusing WCFInterfaces]

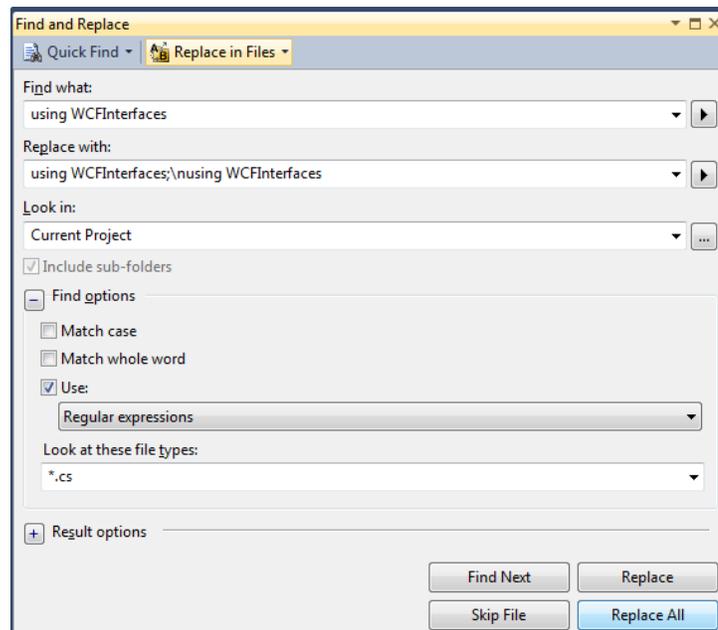


Figura 49 FBSServicios - Modificar importación de WCFInterfaces

2.4.2.11. Excluir archivos. Excluir del proyecto los siguientes archivos:

- Core.Utilidades.RaiseException.cs
- FBSServidor/UsernameAssertionLibrary/ServiceUsernameTokenManager.cs

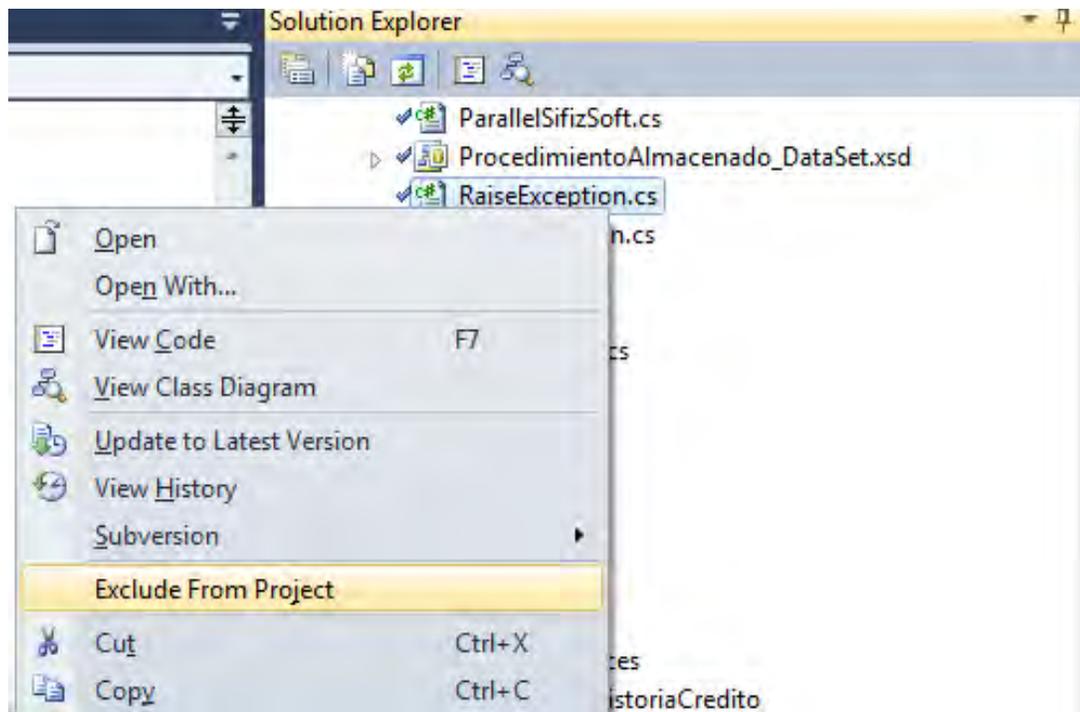


Figura 50 FBSServicios - Excluir archivo RaiseException

2.4.2.12. Agregar ErrorHandler. Incluir en FBSServicios el archivo ErrorHandler.cs que se encuentra en la carpeta Recursos/FBSServicios/ErrorHandler.cs

2.4.2.13. Compilar proyecto. Compilar el proyecto FBSServicios y corregir errores en caso de ser necesario.

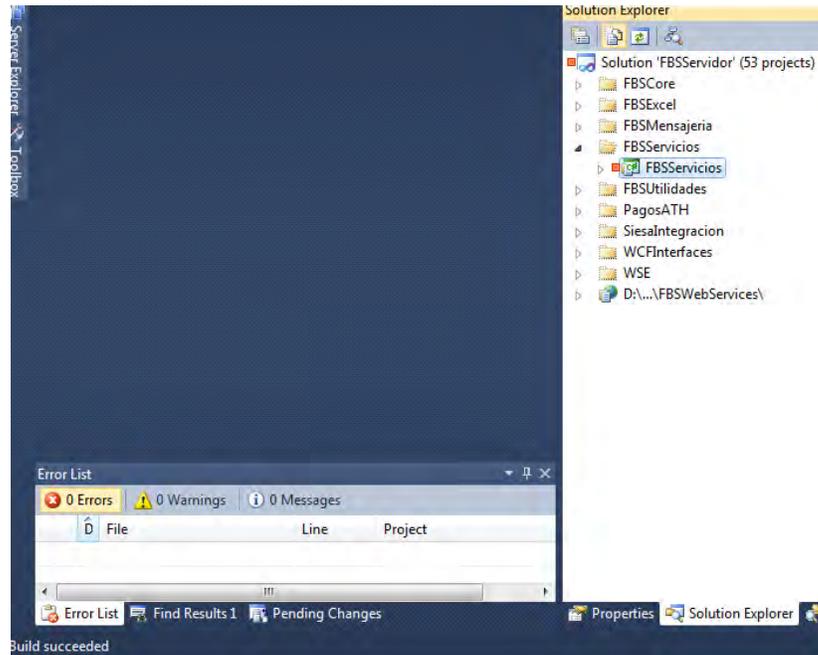


Figura 51 FBSServicios - Compilar proyecto

2.4.3. Convertir el proyecto FBSServicios a un proyecto tipo WCF

2.4.3.1. Copiar FBSServicios a un proyecto tipo WCF. Realizar las siguientes actividades:

- Copiar el proyecto FBSServicios a una carpeta temporal.
- Eliminar el proyecto FBSServicios de la solución FBSServidor (También eliminar la carpeta del proyecto en la ubicación de la solución).
- Agregar un nuevo proyecto

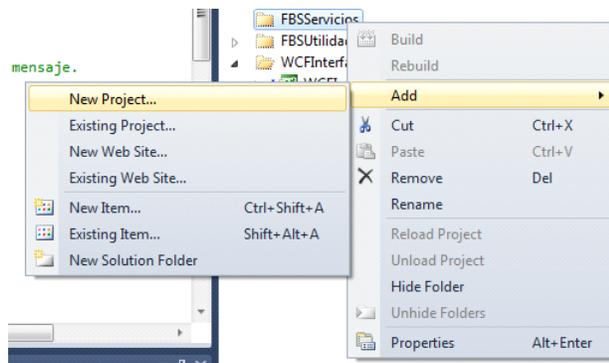


Figura 52 FBSServicios - Agregar nuevo proyecto

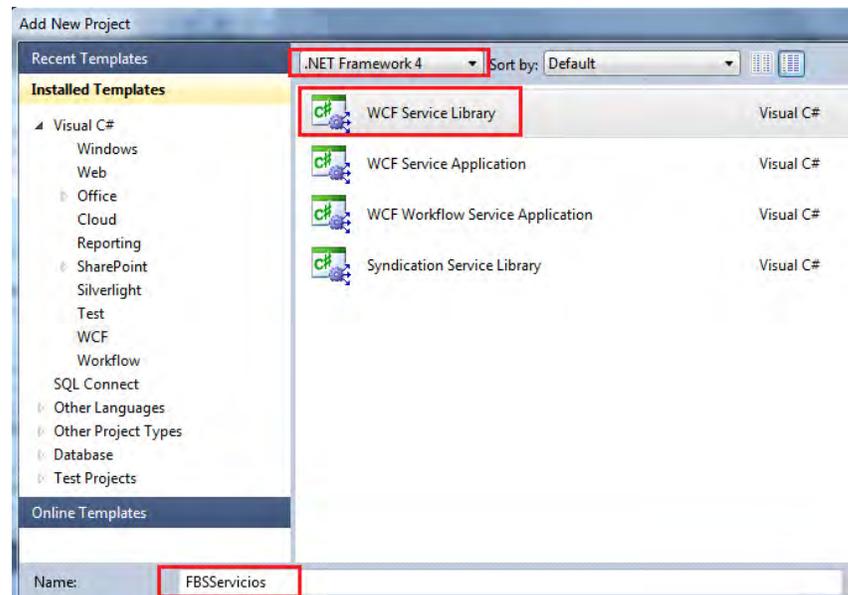


Figura 53 FBSServicios - Elegir tipo de proyecto

- Eliminar los archivos del servicio creado automáticamente (IService1.cs y Service1.cs)

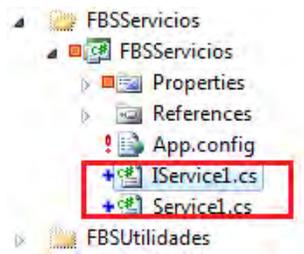


Figura 54 FBSServicios - Eliminar archivos autogenerados

- Activar la opción *Mirar todos los archivos*.



Figura 55 FBSServicios - Activar opción mirar todos los archivos

- Copiar las carpetas de FBSServicios que se copiaron en la carpeta temporal al nuevo proyecto wcf (excepto bin, obj y properties)
- Refrescar la solución y agregar todas las carpetas al proyecto.

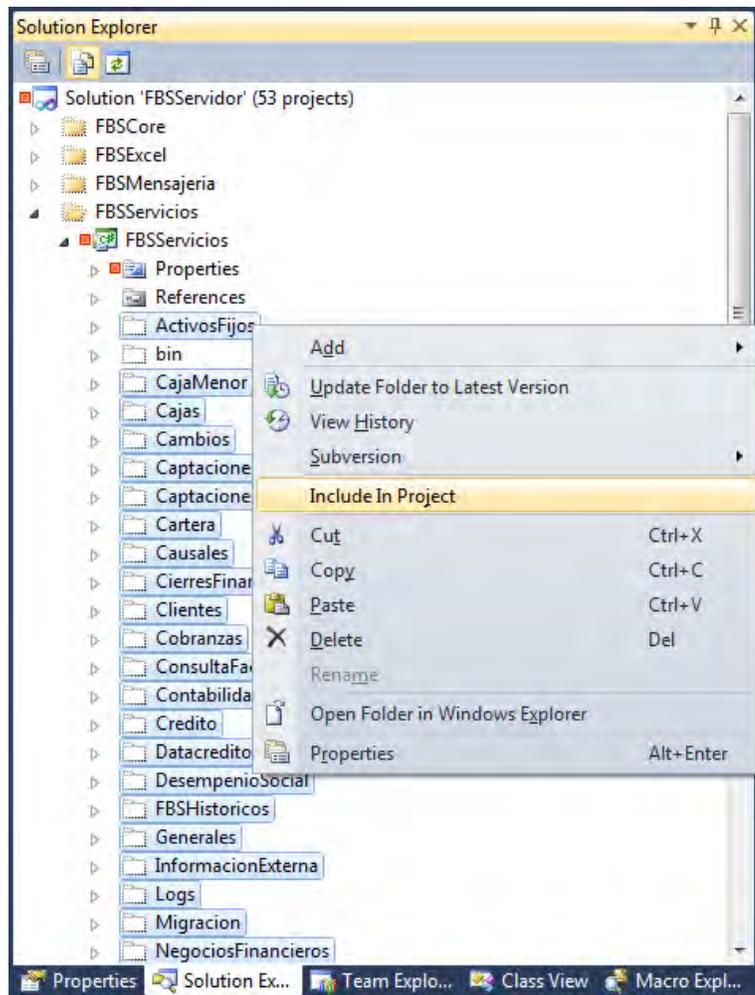


Figura 56 FBSServicios - Incluir carpetas y archivos en nuevo proyecto

2.4.3.2. **Agregar referencias.** Agregar las siguientes referencias al proyecto FBSServicios

- Core/*
- FBSMensajeria
- FBSUtilidades
- ICSharpCode.SharpZipLib
- Microsoft.Practices.EnterpriseLibrary.Common
- Microsoft.Practices.EnterpriseLibrary.Data

- Microsoft.Practices.EnterpriseLibrary.ExceptionHandling
- Microsoft.Web.Services3 (está en C:\Program Files (x86)\Microsoft WSE\v3.0)
- System.Configuration
- System.Configuration.Install
- System.Data.DataSetExtensions
- System.IdentityModel
- System.Messaging
- System.Runtime.Serialization
- System.ServiceModel
- System.ServiceProcess
- System.Transactions
- System.Web (En caso de que no aparezca en el listado, verificar que el proyecto esté en .NET Framework 4.0, NO .NET Framework 4.0 Client Profile)
- System.Web.Services
- System.Xml
- UsernameAssertionLibrary
- Utilidades
- WCFExtras
- WCFInterfaces
- SiesaIntegracion

Excluir del proyecto los archivos que se encuentran incluidos en la siguiente tabla:

Tabla 2 FBSServicios - Archivos a excluir del proyecto

MODULO	ARCHIVO
Cartera	TipoPrestamoComponenteCarteraPorcentajeWS.cs
Contabilidades	EstadoFinancieroWS.cs
Datacredito	AlertaWS.cs
Obligaciones	TipoObligacionComponente_ObligacionWS.cs
Personas	PersonaOrganizacionGrupoResultadosProyectoWS.cs

- Buscar y reemplazar de acuerdo con los siguientes datos

Tabla 3 FBSServicios - Relación buscar y reemplazar

Buscar	Reemplazar por
using System.Web.Services;	//using System.Web.Services;
using System.Web.Services.Protocols;	//using System.Web.Services.Protocols;
using Microsoft.Web.Services3;	//using Microsoft.Web.Services3;

Ejemplo de búsqueda y reemplazo

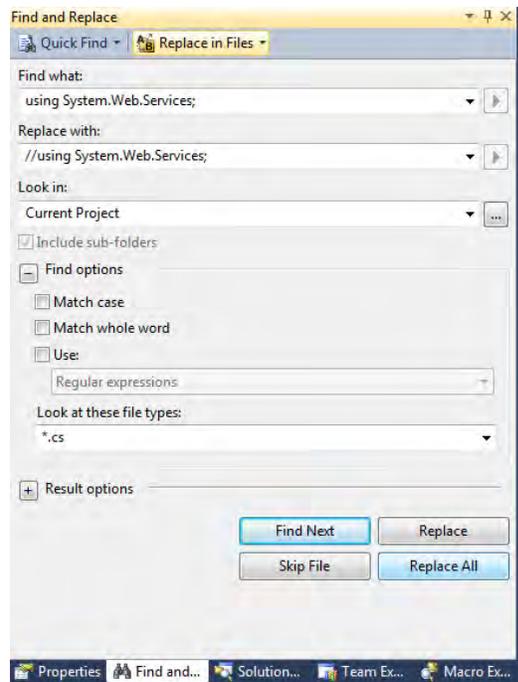


Figura 57 FBSServicios - Ejemplo de búsqueda y reemplazo

2.4.3.3. Revert a AssemblyInfo.cs. Hacer revert a AssemblyInfo.cs (En properties de FBSServicios).

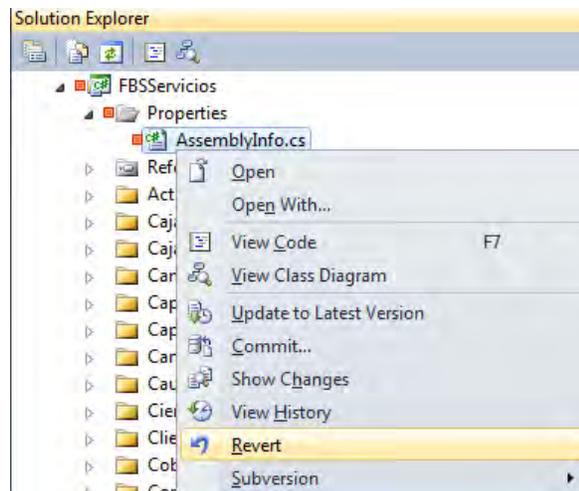


Figura 58 FBSServicios - Revert a AssemblyInfo.cs

2.4.3.4. Editar settings de FBSServidor. Modificar las settings del proyecto FBSServidor.



Figura 59 FBSServicios - Modificar Settings

Agregar una setting con los siguientes datos:

- Name: FBSServicios_ServicioHistoriaCredito_ServicioHistoriaCredito
- Type: Web Service URL
- Scope: Application
- Value: http://172.24.14.7:8080/dhws3/services/DH2ClientesService

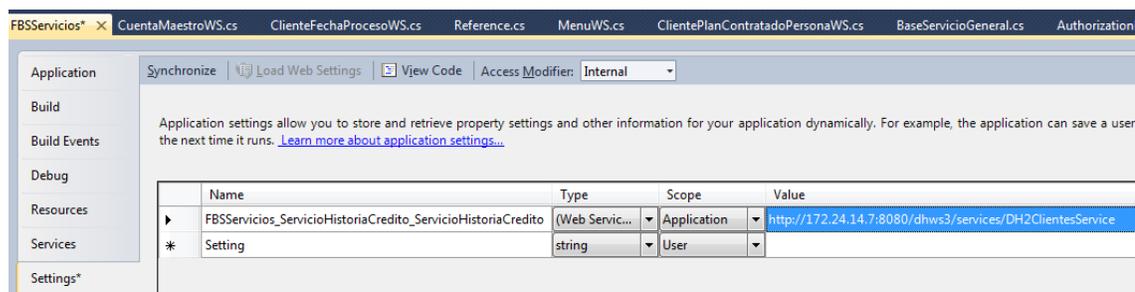
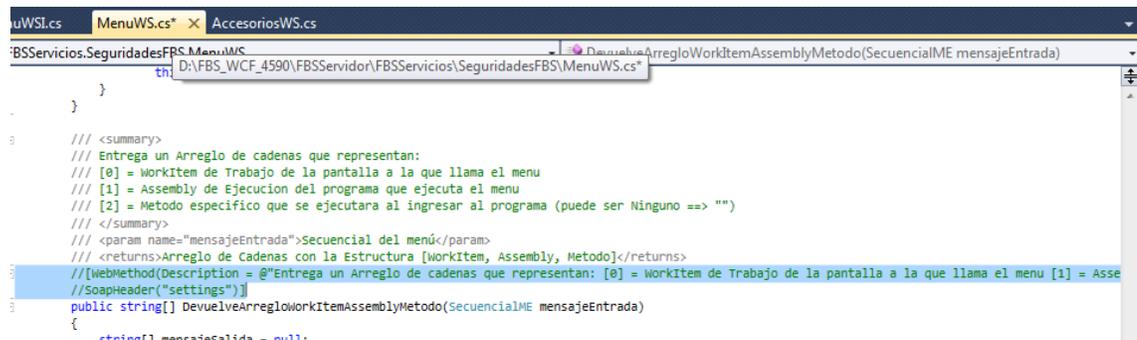


Figura 60 FBSServicios - Agregar setting

2.4.3.5. Ajustes manuales. Comentar la zona marcada en la siguiente imagen para el archivo FBSServicios/SeguridadesFBS/MenuWS.cs.



```
uWSI.cs  MenuWS.cs*  AccesoriosWS.cs
BSServicios.SeguridadesFBS.MenuWS
D:\FBS_WCF_4590\FBSServidor\FBSServicios\SeguridadesFBS\MenuWS.cs*
}
}
}
3  /// <summary>
4  /// Entrega un Arreglo de cadenas que representan:
5  /// [0] = WorkItem de Trabajo de la pantalla a la que llama el menu
6  /// [1] = Assembly de Ejecucion del programa que ejecuta el menu
7  /// [2] = Metodo especifico que se ejecutara al ingresar al programa (puede ser Ninguno ==> "")
8  /// </summary>
9  /// <param name="mensajeEntrada">Secuencial del menú</param>
10 /// <returns>Arreglo de Cadenas con la Estructura [WorkItem, Assembly, Metodo]</returns>
11 /// [WebMethod(Description = @"Entrega un Arreglo de cadenas que representan: [0] = WorkItem de Trabajo de la pantalla a la que llama el menu [1] = Asse
12 /// SoapHeader("settings")]
13 public string[] DevuelveArregloWorkItemAssemblyMetodo(SecuencialME mensajeEntrada)
14 {
15     return mensajeCalida - null;
16 }
```

Figura 61 FBSServicios - Comentar cabeceras de SeguridadesFBS/MenuWS

- Agregar using de System.Web.Services.Protocols; en los archivos:
 - FBSServicios/CaptacionesVista/CuentaMaestroWS.cs
 - FBSServicios/CierresFinancieros/ClienteFechaProcesoWS.cs

2.4.3.6. Buscar y reemplazar. Buscar: [System.Web.HttpContext.Current.User = principal;]

Reemplazar: [] (vacío)

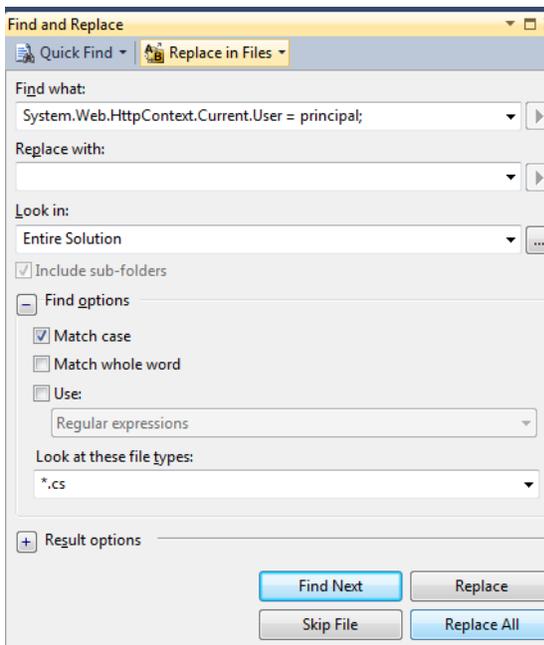


Figura 62 FBSServicios - Eliminar System.Web.HttpContext.Current.User

2.4.3.7. Compilar FBSServicios y corregir errores. Compilar el proyecto FBSServicios y corregir errores en caso de ser necesario

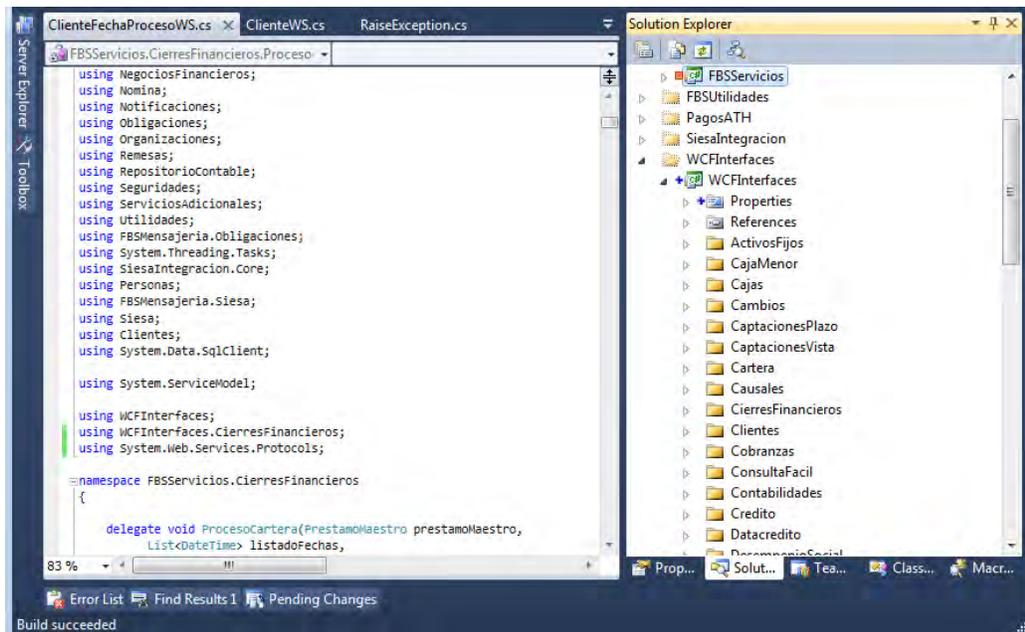


Figura 63 FBSServicios - Compilar proyecto

2.4.4. Modificar proyecto FBSMensajeria

2.4.4.1. Agregar usings necesarios. Buscar y reemplazar de acuerdo con los siguientes datos:

Buscar: [^namespace]

Reemplazar por: [using System.Runtime.Serialization;\n\nnamespace]

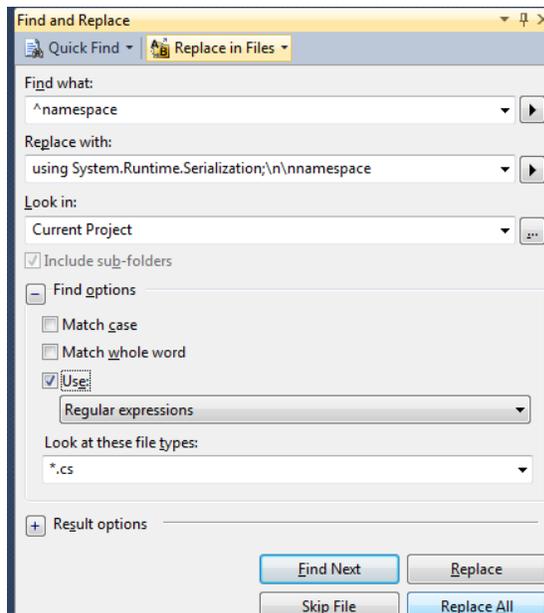


Figura 64 FBSMensajeria - Importar System.Runtime.Serialization

2.4.4.2. Agregar librería a las referencias del proyecto. Agregarla librería System.Runtime.Serialization.

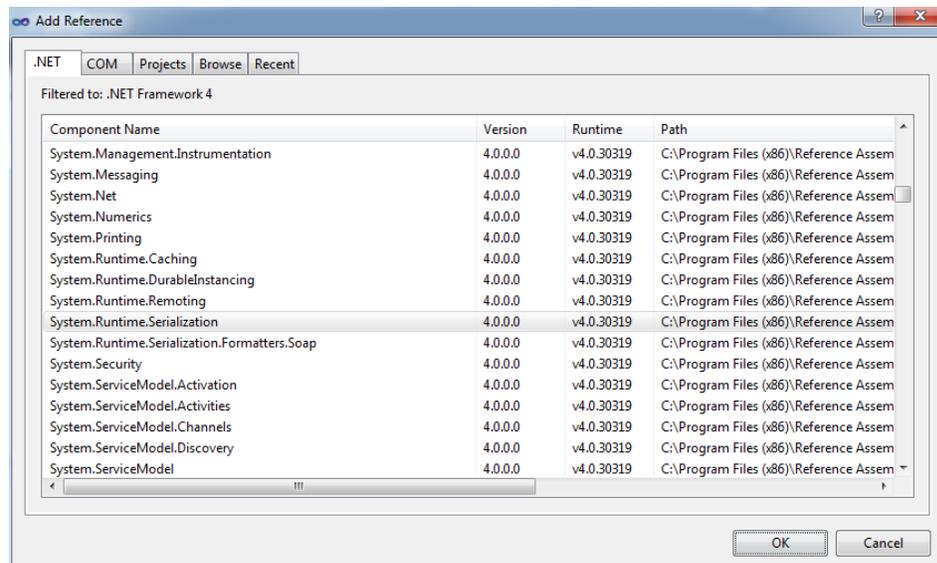


Figura 65 FBSMensajeria - Agregar librería System.Runtime.Serialization

2.4.4.3. Ejecutar macro FBSMensajeria_AgregarEncabezados. Ejecutar las siguientes actividades.

- Abrir algún archivo de FBSMensajeria
- Abrir Macro FBSMensajeria_AgregarEncabezados
- Dentro del proyecto del macro, abrir el archivo RecorrerProyecto
- Ubicarse en el método ProcesarProyecto() y ejecutar (F5)

Tiempo aproximado de ejecución: 30 minutos

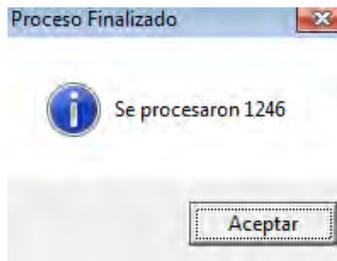


Figura 66 FBSMensajeria - Ejecución FBSMensajeria_AgregarEncabezados

2.4.4.4. Compilar proyecto. Compilar FBSMensajeria y corregir errores.

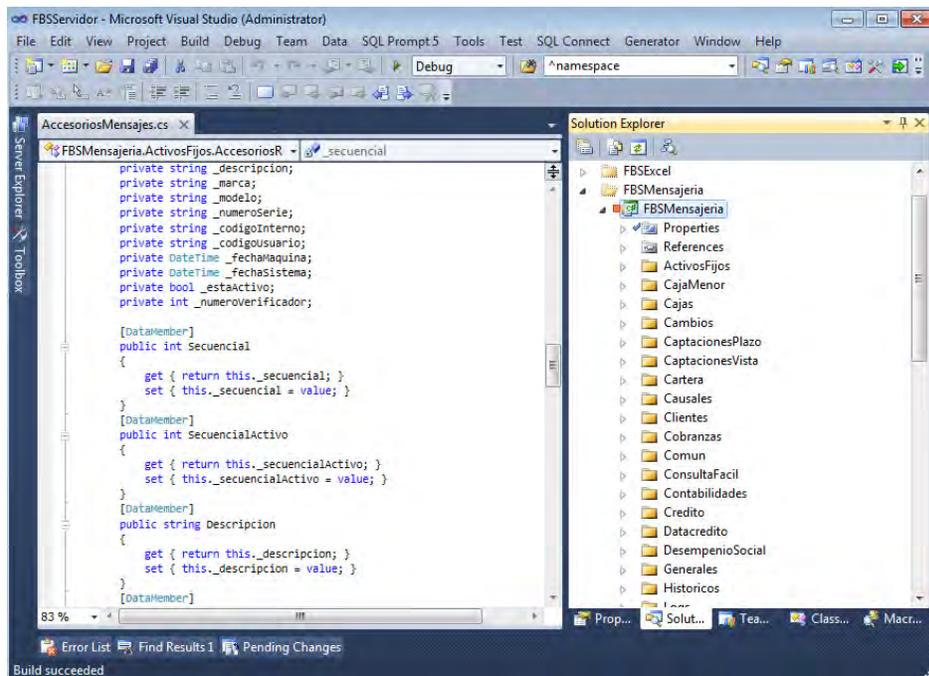


Figura 67 FBSMensajeria - Compilar proyecto

2.4.5. Crear web site

2.4.5.1. Agregar web site a la solución. En la solución, agregar nuevo sitio web.

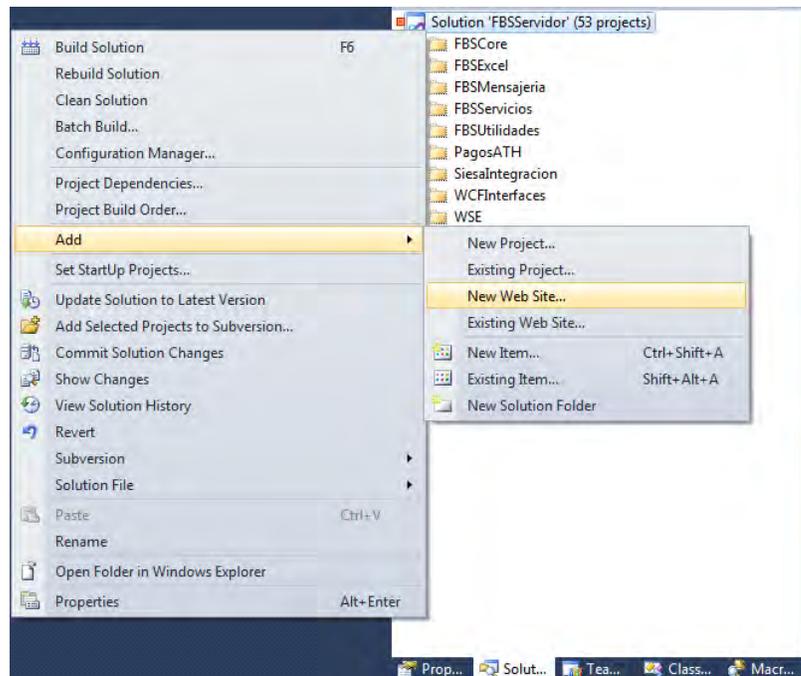


Figura 68 Web site - Nuevo

- Ingresar la ruta de almacenamiento y nombre del proyecto

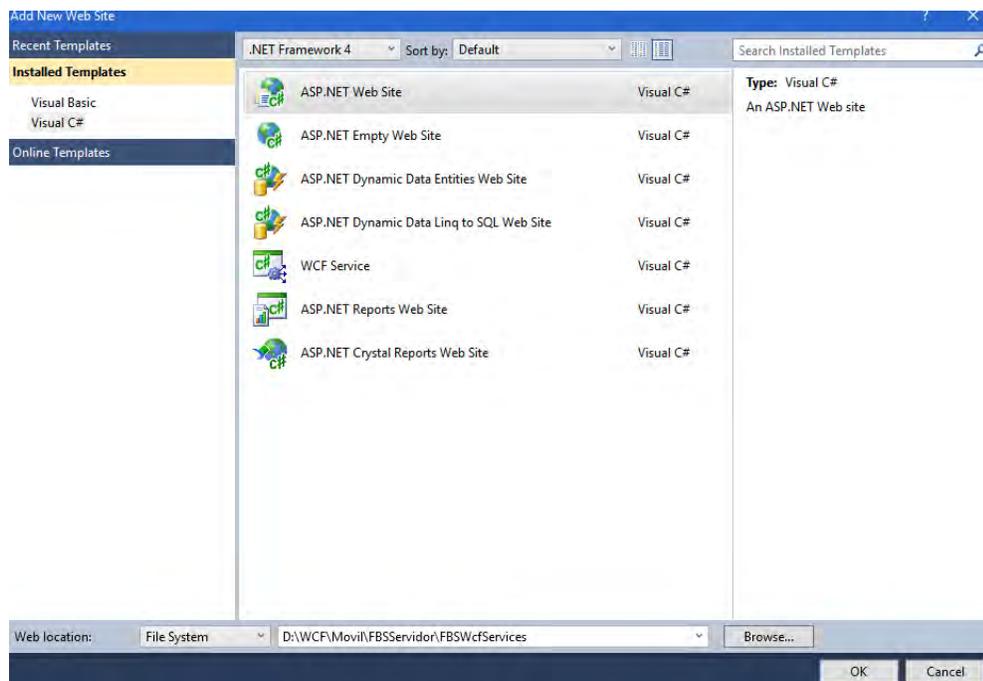


Figura 69 Web site - Ruta de creación

- Web site creado

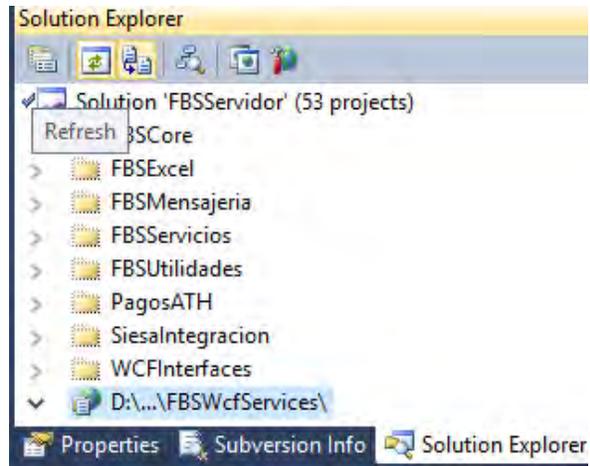


Figura 70 Web site - Creación terminada

- **Borrar los archivos:**
 - App_Code/IService.cs
 - App_Code/Service.cs
 - Service.svc

2.4.5.2. Copiar FBSWebServices. Copiar las carpetas de FBSWebServices a una carpeta temporal, excepto las carpetas: App_Code, Bin, y scripts.

2.4.5.3. Renombrar archivos utilizando el programa Rename-it. Abrir el programa Rename-It! y presionar el botón para agregar carpetas.



Figura 71 Web site - Agregar carpetas para renombrar

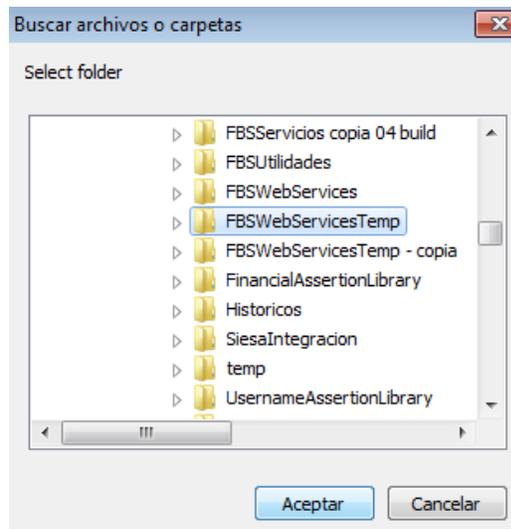


Figura 72 Web site - Carpeta temporal

- Asegurarse de tener marcada la zona de *extensión*.

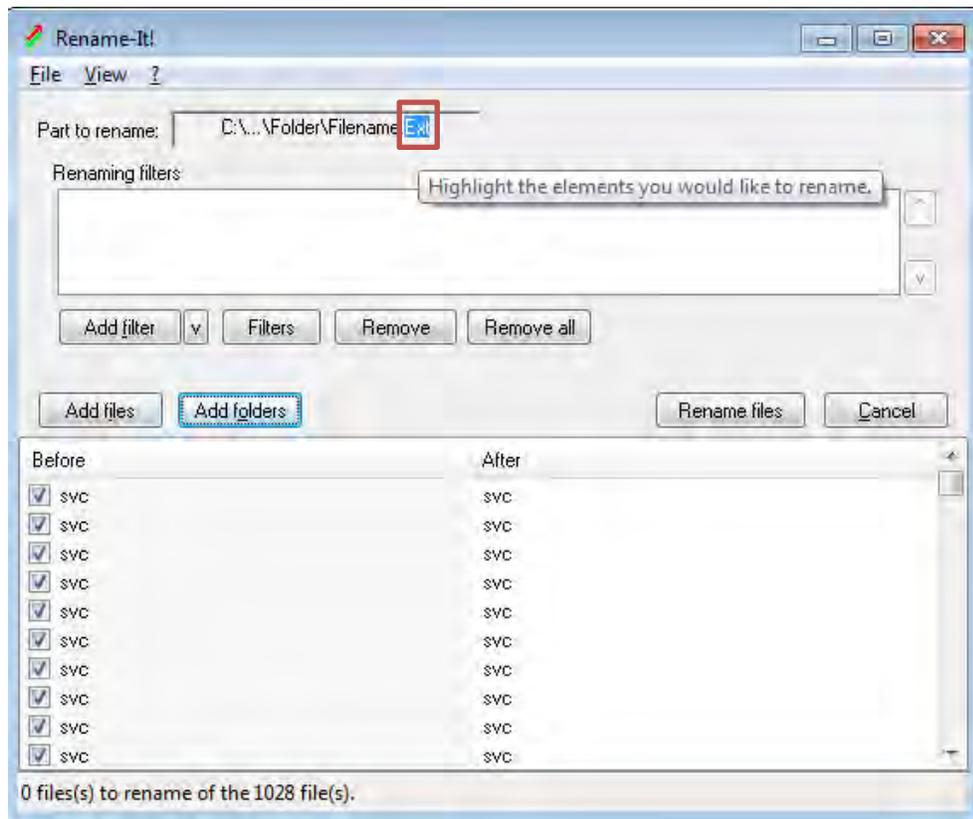


Figura 73 Web site - Selección de parte para renombrar

- Agregar Filtro

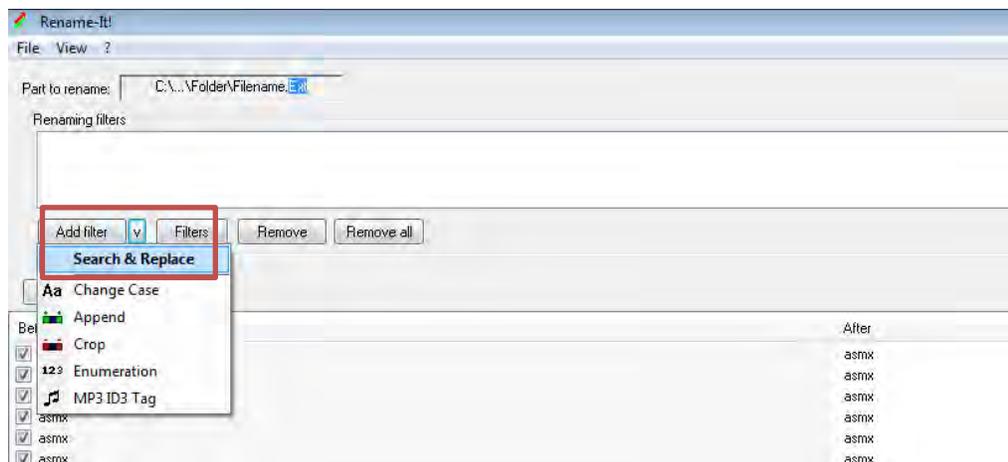


Figura 74 Web site - Agregar filtro search & replace

- Ingresar la nueva extensión de los archivos (svc)

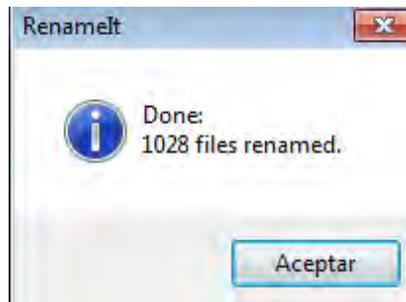


Figura 77 Web site - Renombrado terminado

- Una vez realizado este proceso los nombres de los archivos se verán así:

Nombre	Fecha de modifica...	Tipo	Tamaño
AccesoriosWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
Activo_BajaWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
Activo_FacturaCotizacionWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
Activo_PersonalizadoWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
ActivoComprobanteCtablePagoWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
ActivoOficinaProrateoWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
ActivosFijosReportesWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
ActivoWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB
CalculoWS.svc	15/05/2015 03:36 ...	WCF Web Service	1 KB

Figura 78 Web site - Resultado de renombramiento

2.4.5.4. Cargar carpetas en FBSWcfServices. Copiar carpetas a la carpeta del proyecto FBSWcfServices.

Nombre	Fecha de modifica...	Tipo
ActivosFijos	11/12/2015 6:33 p...	Carpeta de archivos
App_Code	8/01/2016 5:13 p. m.	Carpeta de archivos
Bin	15/12/2015 11:04 a...	Carpeta de archivos
CajaMenor	11/12/2015 6:33 p...	Carpeta de archivos
Cajas	11/12/2015 6:33 p...	Carpeta de archivos
Calidad	11/12/2015 6:33 p...	Carpeta de archivos
Cambios	11/12/2015 6:33 p...	Carpeta de archivos
CaptacionesPlazo	11/12/2015 6:33 p...	Carpeta de archivos
CaptacionesVista	11/12/2015 6:33 p...	Carpeta de archivos
Cartera	11/12/2015 6:33 p...	Carpeta de archivos
Causales	11/12/2015 6:33 p...	Carpeta de archivos

Figura 79 Web site - Copiar carpetas renombradas

- Copiar carpetas a la carpeta del proyecto FBSWcfServices
- En Visual Studio, seleccionar el proyecto FBSWcfServices y presionar actualizar.
- El resultado es:

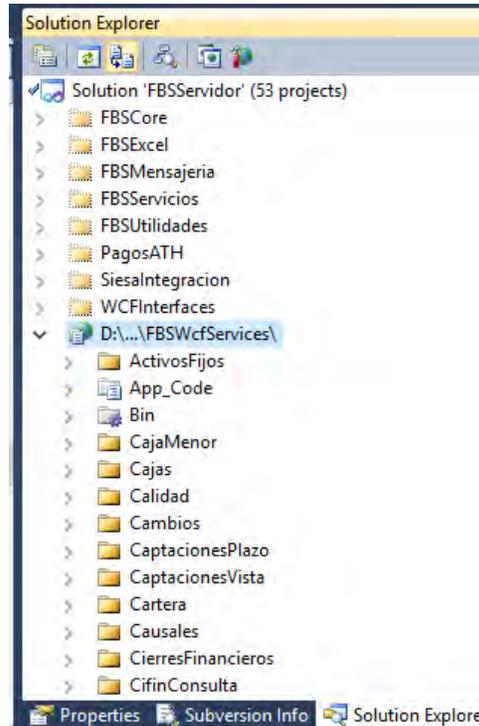


Figura 80 Web site - Estructura con las nuevas carpetas

2.4.5.5. Modificar código de archivos svc. Realizar las siguientes actividades:

- Abrir un archivo de FBSWcfServices
- Buscar y reemplazar en los archivos del proyecto

Buscar: [`<%\@ WebService Language="C#" Class="{([\^"]*)}`]

Reemplazar por: [`<%@ ServiceHost Language="C#" Debug="true" Service="\1`]

- Indicar el tipo de extension *.svc

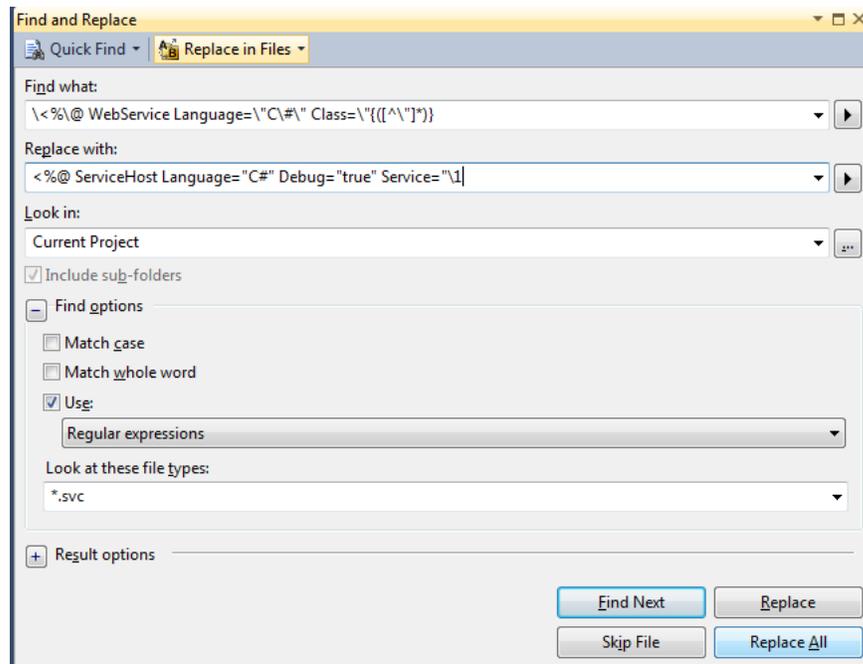


Figura 81 Web site - Modificar código de archivos svc

- Confirmar reemplazo

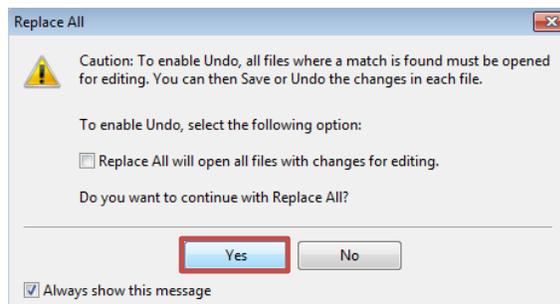


Figura 82 Web site - Confirmar reemplazo

- Proceso terminado

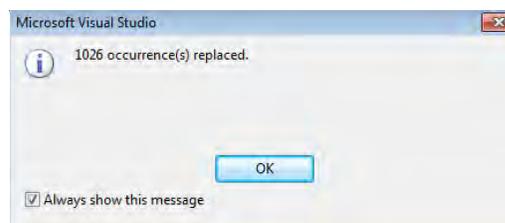


Figura 83 Web site - Reemplazo terminado

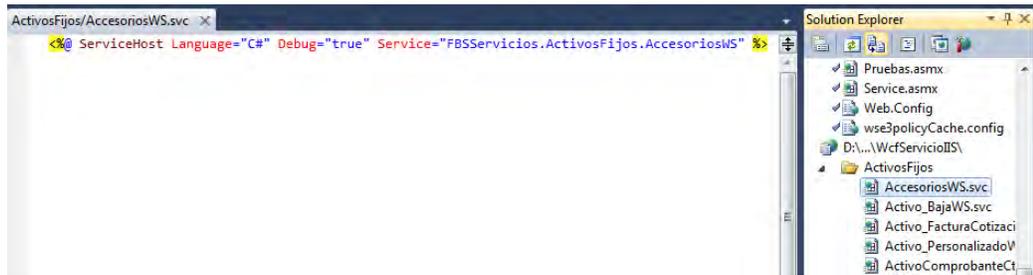


Figura 84 Web site - Ejemplo código de archivo svc

2.4.5.6. Agregar servicios al Web.config. Abrir un archivo de FBSWcfServices y buscar en sus archivos:

Buscar: [Service="]

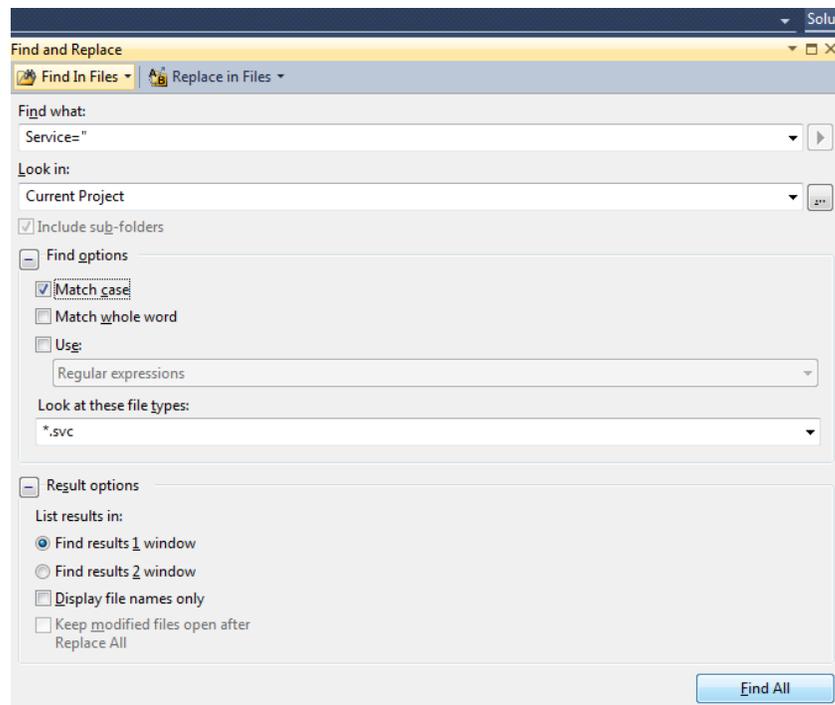


Figura 85 Web site - Buscar en todo el proyecto

- Resultado de la búsqueda:

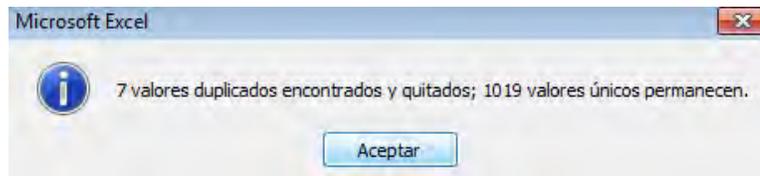


Figura 88 Excel - Duplicados eliminados

- Seleccionar todos los datos

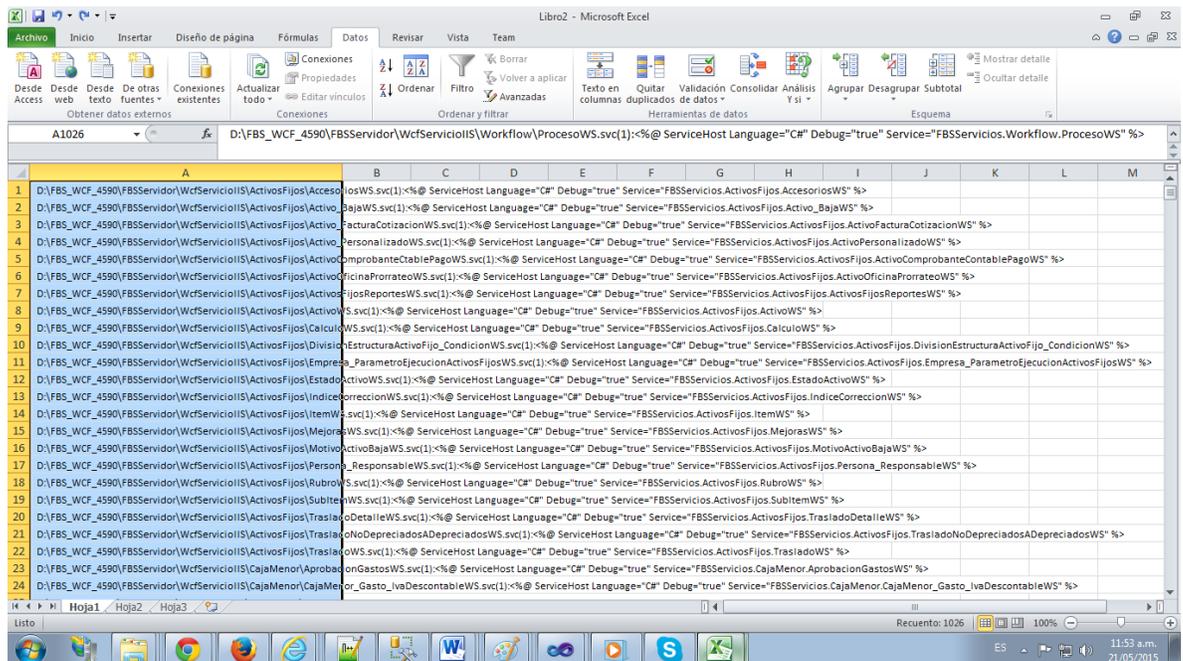


Figura 89 Excel - Seleccionar todos los datos

- En Excel ir al menú: Datos > Texto en columnas

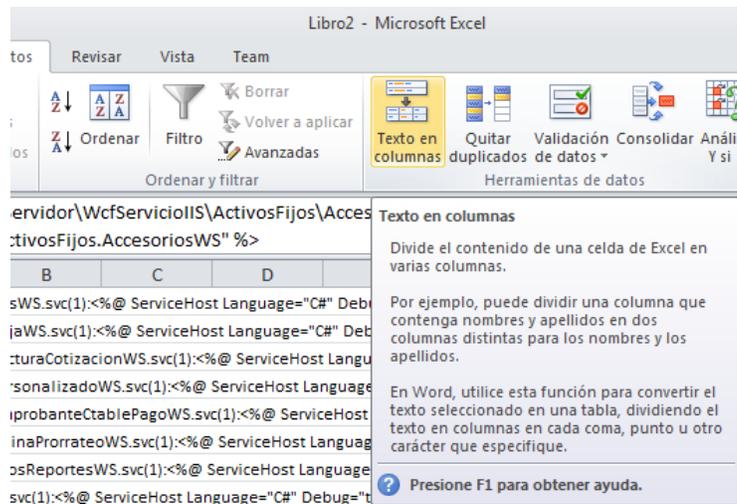


Figura 90 Excel - Datos como texto en columnas

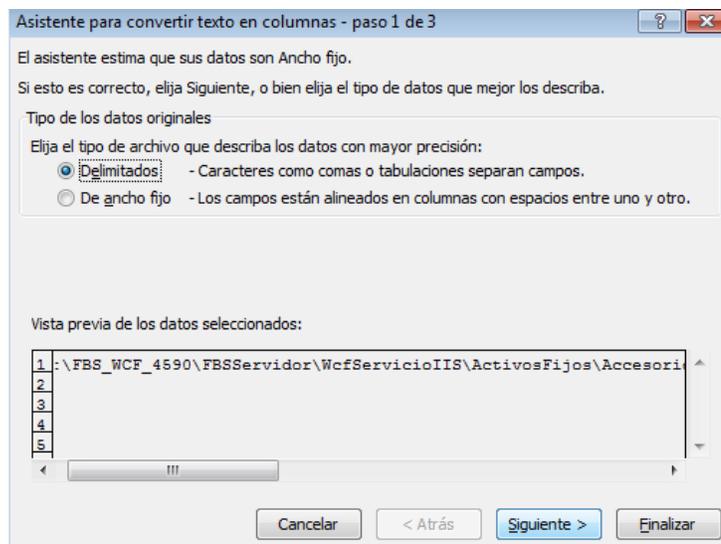


Figura 91 Excel - Elegir tipo de archivo

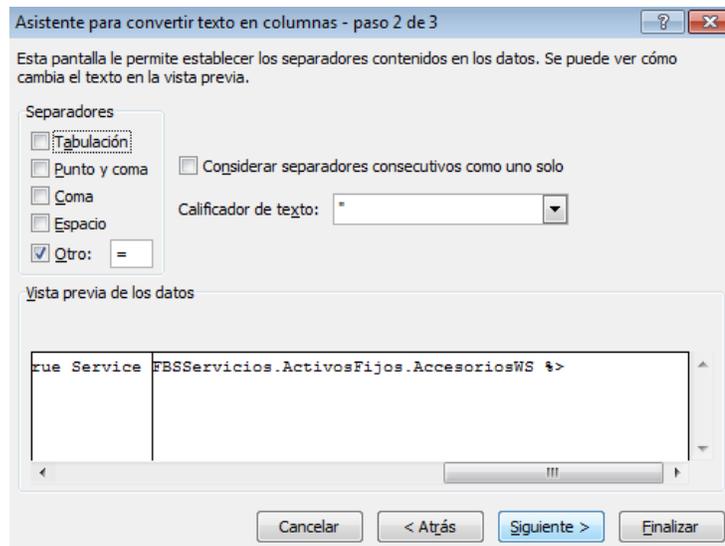


Figura 92 Excel - Elegir carácter separador

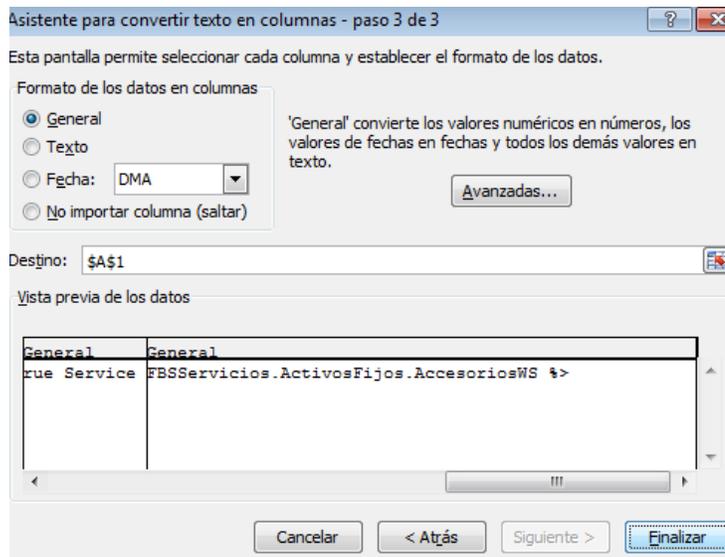


Figura 93 Excel - Elegir formato de las columnas

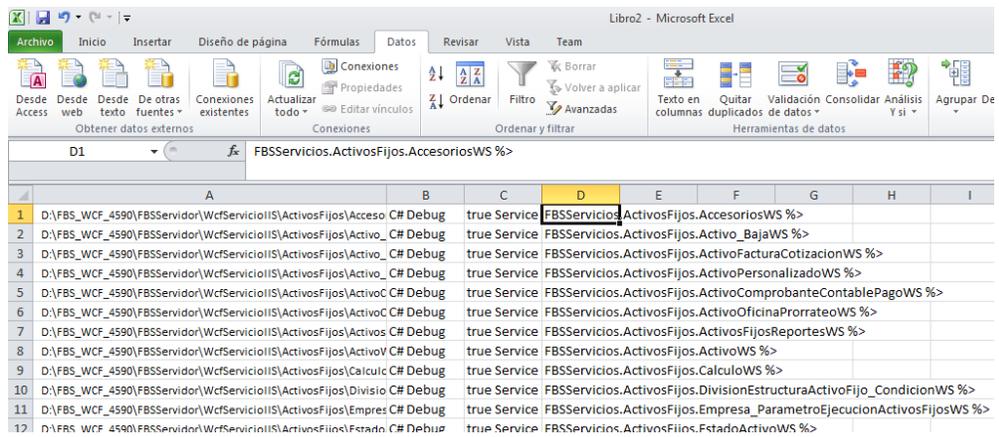


Figura 94 Excel - Borrado de columnas

- Borrar las columnas A, B y C.
- Seleccionar todos los datos.
- Buscar y reemplazar.

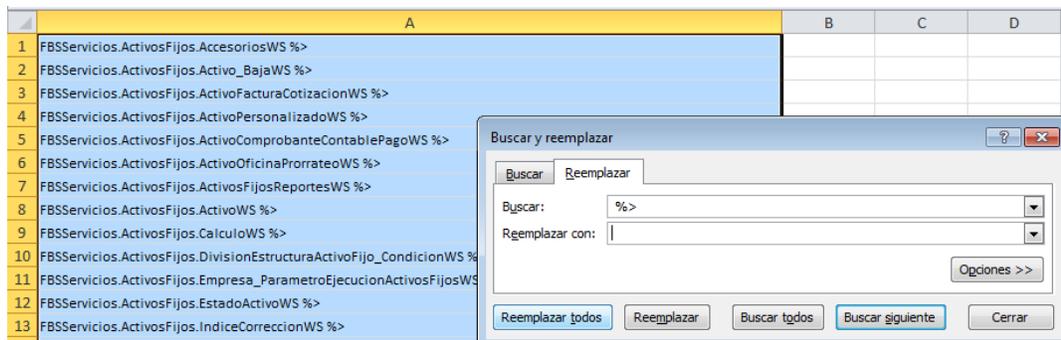


Figura 95 Excel - Eliminar carácter especial

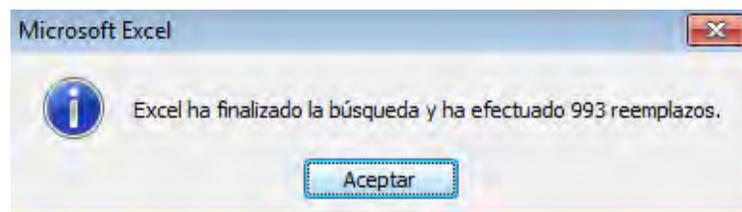


Figura 96 Excel - Eliminación terminada

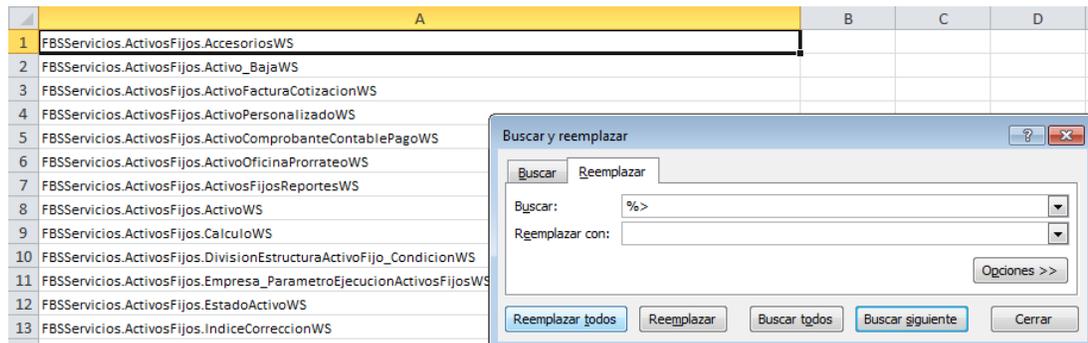


Figura 97 Excel - Eliminar carácter XML

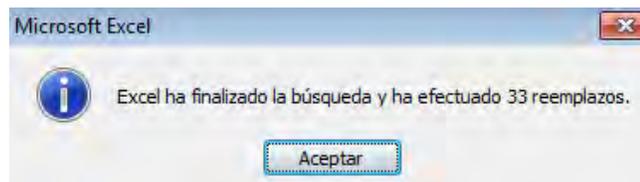


Figura 98 Excel - Carácter XML eliminado

- Sumar los dos resultados para confirmar que el reemplazo se hizo en todas las filas

	A
1009	FBSServicios.Soporte.SolicitudSoporteEscalaUsuarioWS
1010	FBSServicios.Soporte.SolicitudSoporteSeguimientoWS
1011	FBSServicios.Soporte.SolicitudSoporteWS
1012	FBSServicios.Soporte.TipoSolicitudSoporteWS
1013	FBSServicios.Soporte.Usuario_SolicitudSoporteEncargadoWS
1014	FBSServicios.Tesoreria.DocumentoSeguridadWS
1015	FBSServicios.Tesoreria.EstadoDocumentoSeguridadWS
1016	FBSServicios.Tesoreria.TipoDocumentoSeguridadWS
1017	FBSServicios.TransaccionesEnLinea.DistribuidorWS
1018	FBSServicios.TransaccionesEnLinea.FormularioLicitoWS
1019	FBSServicios.Utilidades.InformacionReporteWS
1020	FBSServicios.Utilidades.TipoAccionLogWS
1021	FBSServicios.Utilidades.UtilidadesWS
1022	FBSServicios.Workflow.EtapaRetornoWS
1023	FBSServicios.Workflow.EtapaWS
1024	FBSServicios.Workflow.GrupoTrabajoUsuarioWS
1025	FBSServicios.Workflow.GrupoTrabajoWS
1026	FBSServicios.Workflow.ProcesoWS

Figura 99 Excel – Verificación

- Realizar una copia del Web.config

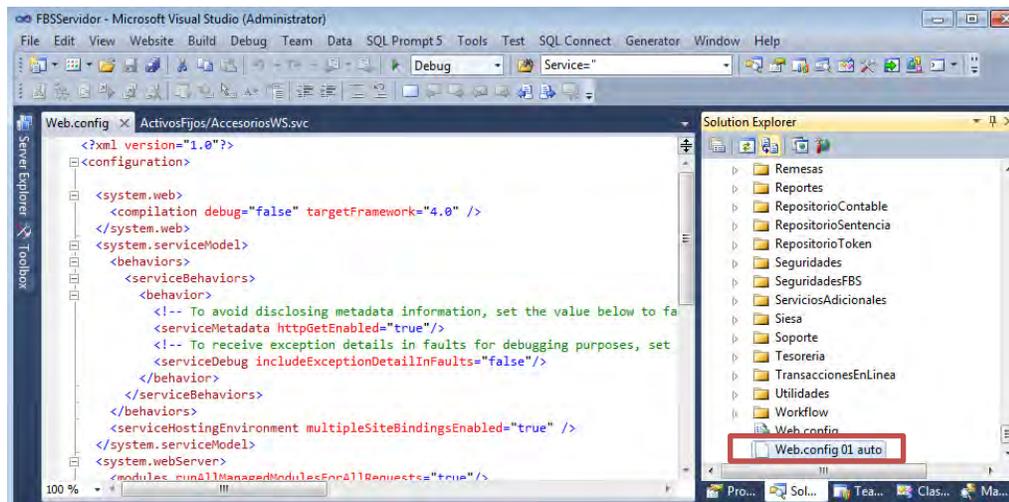


Figura 100 Web.config - Backup del archive Web.config

- Reemplazar el contenido de Web.config por los datos que se encuentran en Excel

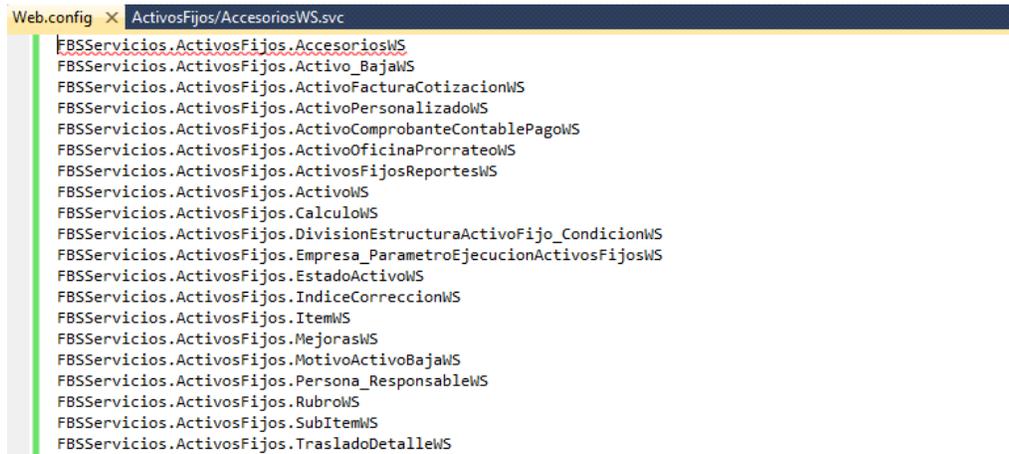


Figura 101 Web.config - Reemplazar contenido del Web.config

- En este documento buscar la siguiente expresión regular:

Buscar: `[^([a-zA-Z0-9_]*).{([a-zA-Z0-9_]*)}.{([a-zA-Z0-9_]*)}]`

Reemplazar por:

```
\<service          behaviorConfiguration="FBSServicios.FinancialBehavior"
name="\1.\2.\3"\>\n          \<endpoint          address=""
behaviorConfiguration="BehaviorMessageInspector"          binding="netTcpBinding"
bindingConfiguration="FinancialNetTcpEndpointBinding"
```

```

name="NetTcpBindingEndpoint" contract="WCFInterfaces.\2.\3" \>\n <endpoint
address="mex" binding="mexTcpBinding" bindingConfiguration=""
name="MexTcpBindingEndpoint" contract="IMetadataExchange" \>\n \<host\>\n
\<baseAddresses\>\n \<add baseAddress="net.tcp://localhost/1/2/3.svc"
\>\n \</baseAddresses\>\n \</host\>\n\</service\>\n]

```

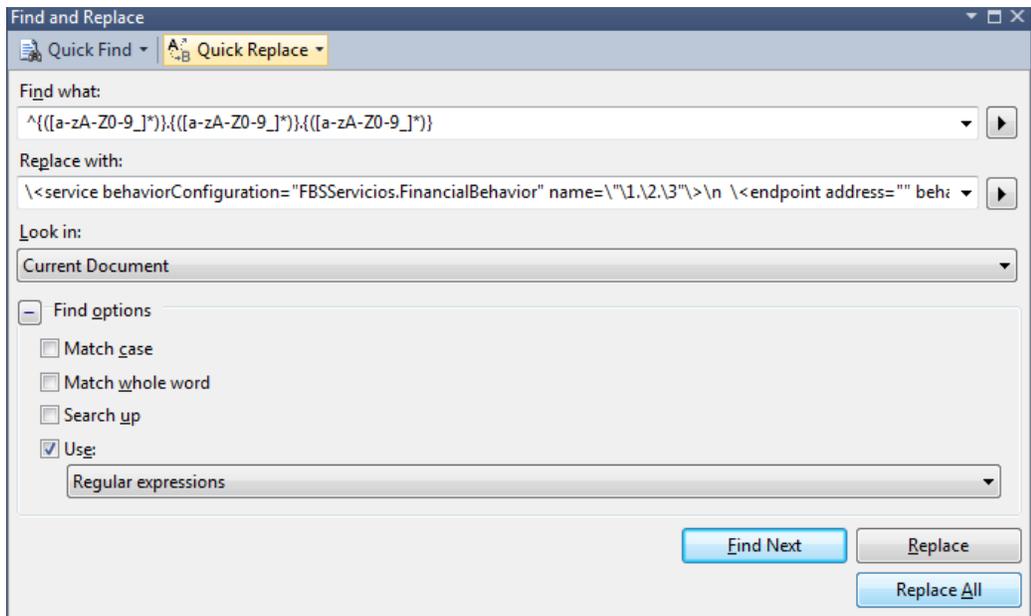


Figura 102 Web.config - Reemplazo con expresión regular

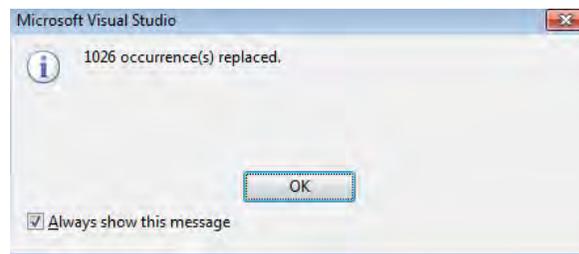


Figura 103 Web.config - Reemplazo terminado

- Así quedan los servicios:

```

Web.config* X ActivosFijos/AccesoriosWS.svc
<service behaviorConfiguration="FBSServicios.FinancialBehavior" name="FBSServicios.ActivosFijos.AccesoriosWS">
  <endpoint address="" behaviorConfiguration="BehaviorMessageInspector" binding="netTcpBinding" bindingConfiguration="FinancialNetTcpEnd
  <endpoint address="mex" binding="mexTcpBinding" bindingConfiguration="" name="MexTcpBindingEndpoint" contract="IMetadataExchange" />
  <host>
    <baseAddresses>
      <add baseAddress="net.tcp://localhost:8523/FBSServicios.ActivosFijos.AccesoriosWS" />
    </baseAddresses>
  </host>
</service>

<service behaviorConfiguration="FBSServicios.FinancialBehavior" name="FBSServicios.ActivosFijos.Activo_Bajaws">
  <endpoint address="" behaviorConfiguration="BehaviorMessageInspector" binding="netTcpBinding" bindingConfiguration="FinancialNetTcpEnd
  <endpoint address="mex" binding="mexTcpBinding" bindingConfiguration="" name="MexTcpBindingEndpoint" contract="IMetadataExchange" />
  <host>
    <baseAddresses>
      <add baseAddress="net.tcp://localhost:8523/FBSServicios.ActivosFijos.Activo_Bajaws" />
    </baseAddresses>
  </host>
</service>

```

Figura 104 Web.config - Ejemplo resultado etiqueta service con contenido

2.4.5.7. Agregar referencias. Agregar referencia a los siguientes proyectos:

- WCFInterfaces
- FBSServicios

2.4.5.8. Compilar Web Site. Compilar web site y corregir errores en caso de ser necesario.

2.4.6. Agregar y configurar aplicación en IIS

2.4.6.1. Agregar aplicación FBSWcfServices. Abrir IIS y agregar aplicación al default web site.

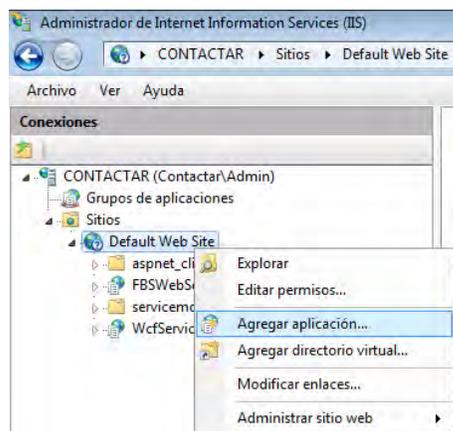


Figura 105 IIS - Agregar aplicación

- Ingresar el alias de la aplicación

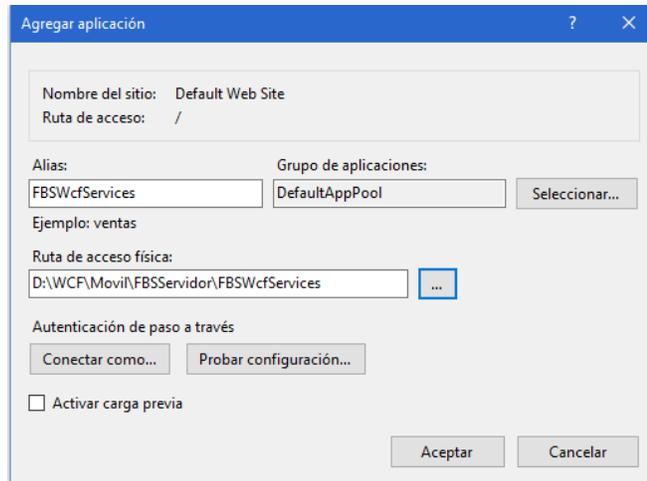


Figura 106 IIS - Ingresar alias de la aplicación

- Elegir la ruta de acceso física.
- Conectar como (el usuario seleccionado debe tener privilegios de acceso a la clave privada del certificado).

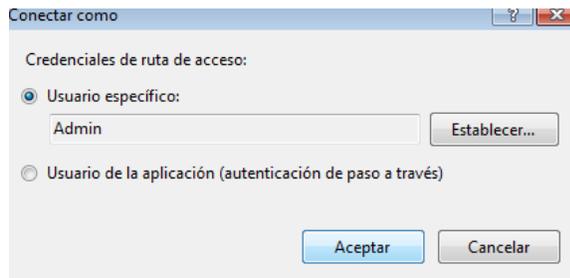


Figura 107 IIS - Seleccionar usuario de conexión

- Probar Conexión.

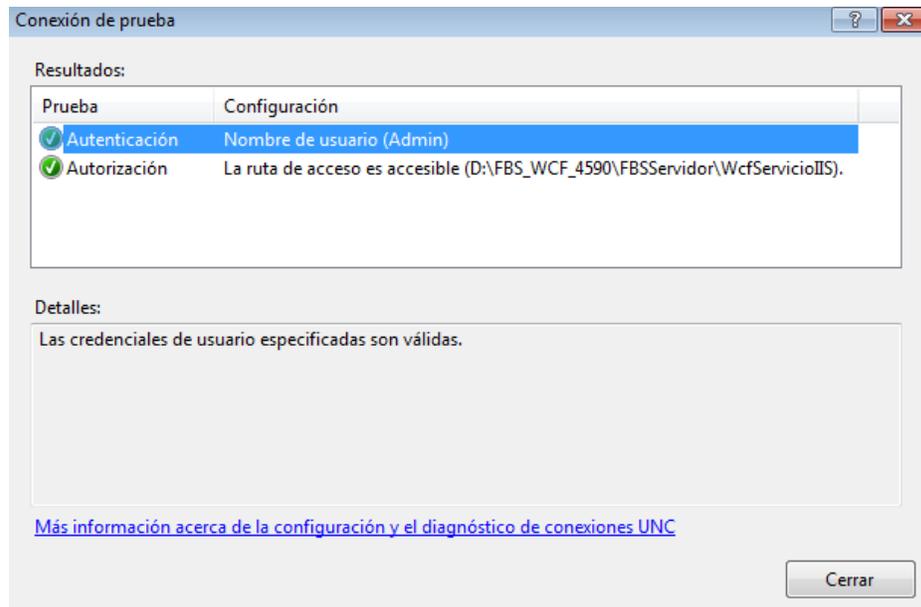


Figura 108 IIS - Prueba de conexión

- Al terminar se mira la nueva aplicación así:

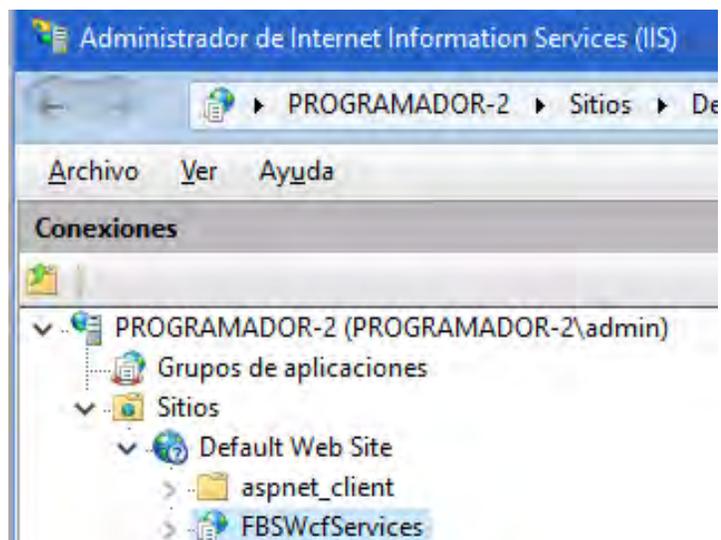


Figura 109 IIS - Nueva aplicación creada

2.4.6.2. Configurar aplicación. Ir a Administrar aplicación > Configuración avanzada.

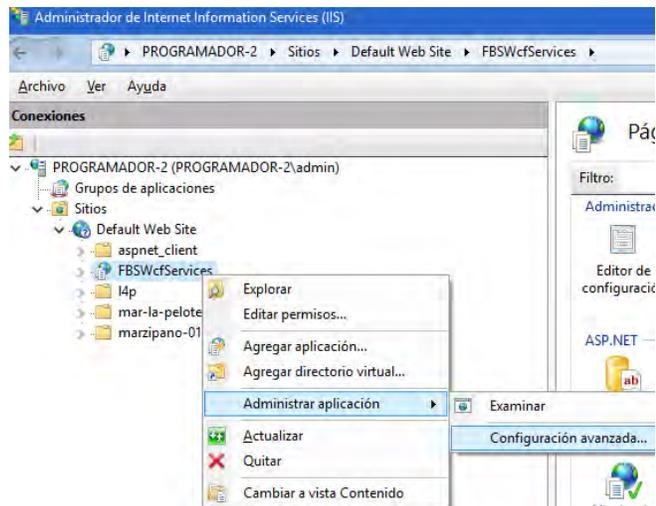


Figura 110 IIS - Configuración avanzada de aplicación

- Así se encuentra

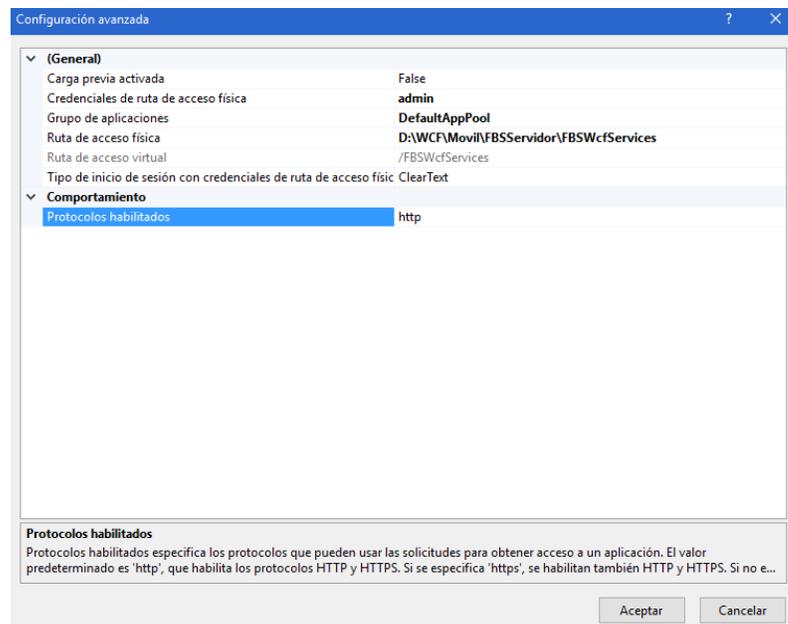


Figura 111 IIS - Estado inicial de configuración

- Agregar el protocolo *net.tcp* y aceptar

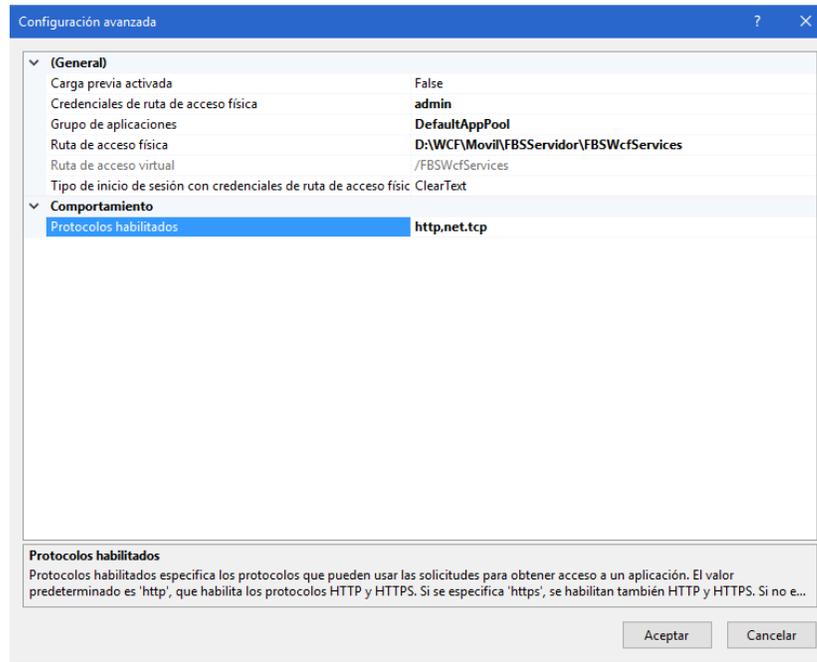


Figura 112 IIS - Protocolo net.tcp agregado a la aplicación

- Habilitar net.tcp en el web site

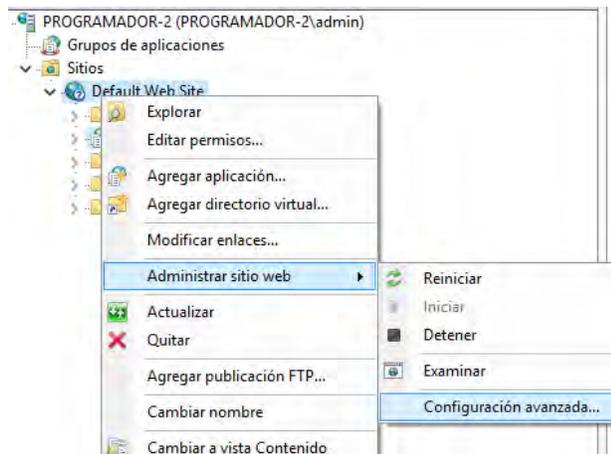


Figura 113 IIS - Abrir administrar sitio web

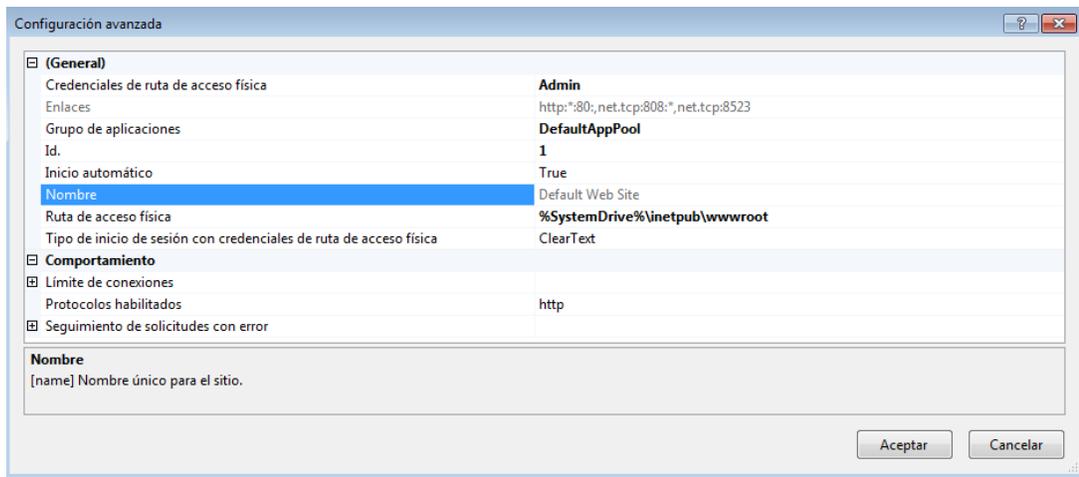


Figura 114 IIS - Configuración original del web site

- Agregar net.tcp a la lista de protocolos habilitados

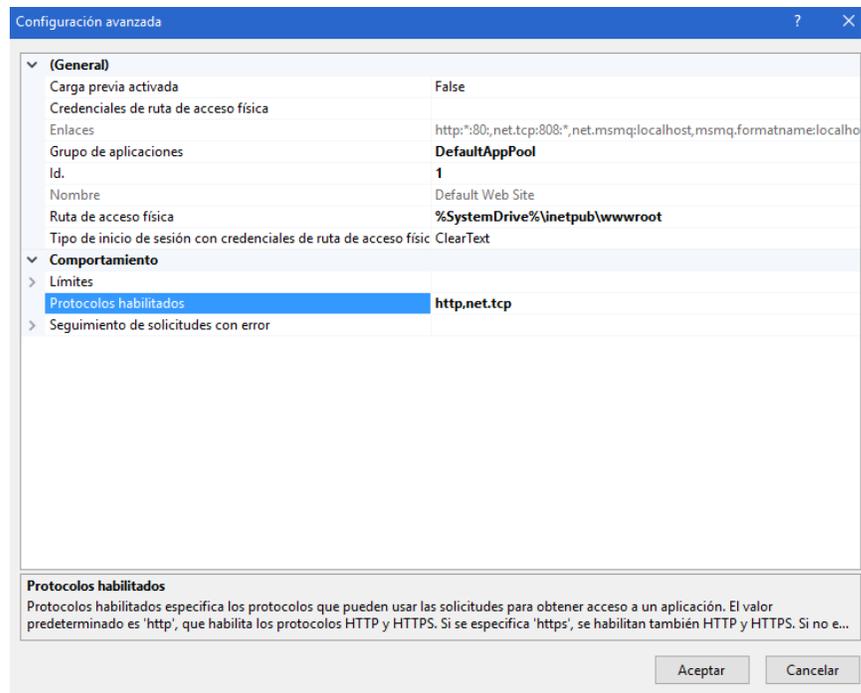


Figura 115 IIS - Protocolo net.tcp agregado al sitio web

- Modifica enlaces del default web site



Figura 116 IIS - Modificar enlaces

- En caso de que no exista, se debe agregar el enlace net.tcp con puerto 808.*

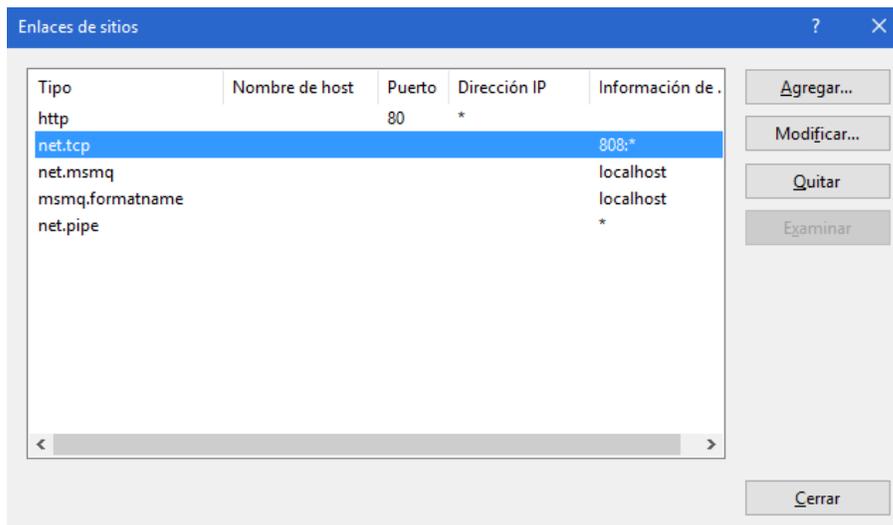


Figura 117 IIS - Enlaces de sitios

2.4.7. Configuraciones extras

2.4.7.1. Convertir XmlDocument a XElement. Convertir XmlDocument a XElement en FBSServicios/Credito/Usuario_OficialDivisionPoIWS.cs.



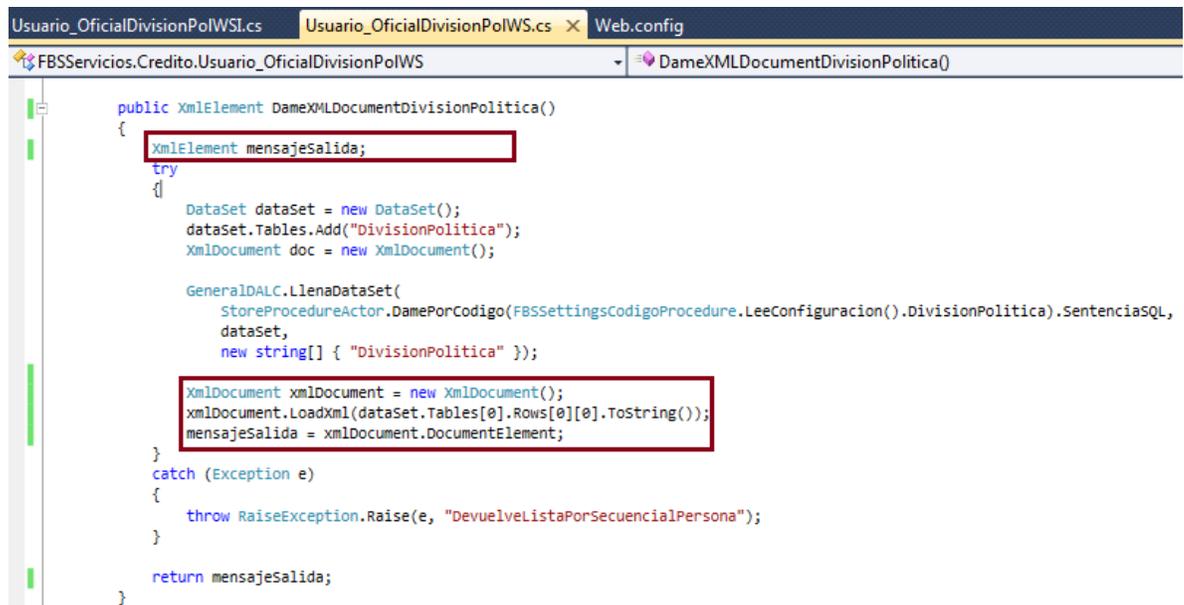
```
public XmlDocument DameXMLDocumentDivisionPolitica()
{
    XmlDocument mensajeSalida = new XmlDocument();
    try
    {
        DataSet dataSet = new DataSet();
        dataSet.Tables.Add("DivisionPolitica");
        XmlDocument doc = new XmlDocument();

        GeneralDALC.LlenaDataSet(
            StoreProcedureActor.DamePorCodigo(FBSSettingsCodigoProcedure.LeeConfiguracion()).DivisionPolitica,
            dataSet,
            new string[] { "DivisionPolitica" });

        mensajeSalida.LoadXml(dataSet.Tables[0].Rows[0][0].ToString());
    }
}
```

Figura 118 Configuraciones extras - Actualizar XmlDocument 1

Pasa a ser:



```
public XElement DameXMLDocumentDivisionPolitica()
{
    XElement mensajeSalida;
    try
    {
        DataSet dataSet = new DataSet();
        dataSet.Tables.Add("DivisionPolitica");
        XmlDocument doc = new XmlDocument();

        GeneralDALC.LlenaDataSet(
            StoreProcedureActor.DamePorCodigo(FBSSettingsCodigoProcedure.LeeConfiguracion()).DivisionPolitica).SentenciaSQL,
            dataSet,
            new string[] { "DivisionPolitica" });

        XmlDocument xmlDocument = new XmlDocument();
        xmlDocument.LoadXml(dataSet.Tables[0].Rows[0][0].ToString());
        mensajeSalida = xmlDocument.DocumentElement;
    }
    catch (Exception e)
    {
        throw RaiseException.Raise(e, "DevuelveListaPorSecuencialPersona");
    }

    return mensajeSalida;
}
```

Figura 119 Configuraciones extras - Convertir a XElement 1

- Actualizar la interfaz correspondiente

```

WCFInterfaces.Credito.IUsuario_OficialDivisionPolWS
DameXMLDocumentDivisionPolitica()

/// <summary>
/// Devuelve un registro de divisiones politicas de acuerdo al codigo de una usuario y secuencia
/// </summary>
/// <param name="mensajeEntrada">Codigo de un usuario oficial</param>
/// <param name="mensajeEntrada">Secuencial de division Politica</param>
/// <returns>Mensaje con la lista de divisiones politicas</returns>
[SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
[FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
[OperationContract(Name = "DameXMLDocumentDivisionPolitica")]
//System.Xml.XmlDocument DameXMLDocumentDivisionPolitica();
public XmlElement DameXMLDocumentDivisionPolitica();
}

```

Figura 120 Configuraciones extras - Actualización de interfaz 1

- Verificar el ingreso al servicio

localhost/FBSWcfServices/Credito/Usuario_OficialDivisionPolWS.svc

Uuario_OficialDivisionPolWS Service

You have created a service.

To test this service, you will need to create a client and use it to call the service. You can do this using the svcutil.exe tool from the command line with the following syntax:

```
svcutil.exe net.tcp://contactar/FBSWcfServices/Credito/Usuario_OficialDivisionPolWS.svc/mex
```

This will generate a configuration file and a code file that contains the client class. Add the two files to your client application and use the generated client class to call the Service. For example:

```

C#
class Test
{
    static void Main()
    {
        Usuario_OficialDivisionPolWSClient client = new Usuario_OficialDivisionPolWSClient();

        // Use the 'client' variable to call operations on the service.

        // Always close the client.
        client.Close();
    }
}

```

Figura 121 Configuraciones extras - Verificación de acceso 1

- Convertir XmlDocument a XElement en FBSServicios/RepositorioToken/TokenWS.cs

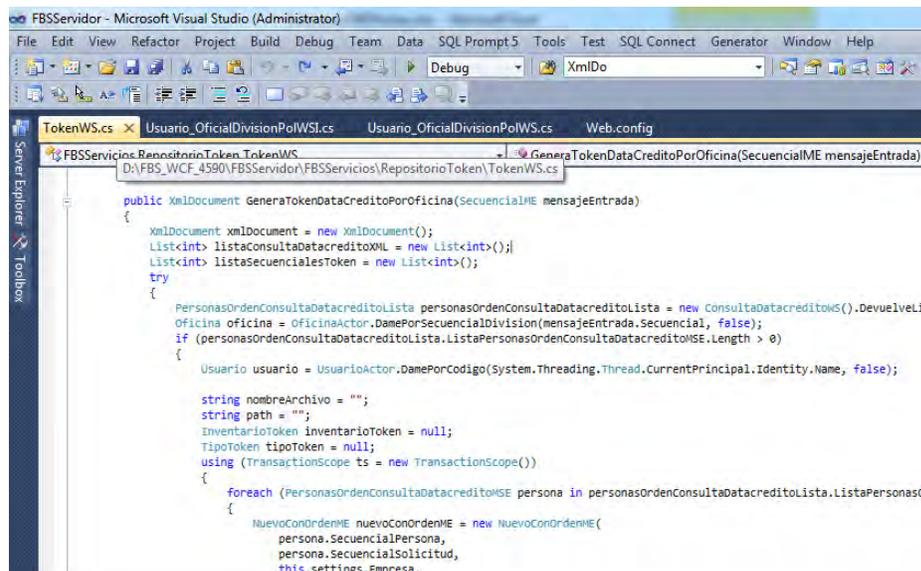


Figura 122 Configuraciones extras – Actualizar XmlDocument 2

- Pasa a ser:

```

public XmlElement GeneraTokenDataCreditoPorOficina(SecuencialME mensajeEntrada)
{
    XmlElement xmlElement;
    XmlDocument xmlDocument = new XmlDocument();
    List<int> listaConsultaDatacreditoXML = new List<int>();
    List<int> listaSecuencialesToken = new List<int>();
    try
    {
        catch
        {
        }
    }
    catch (Exception e)
    {
        throw RaiseException.Raise(e, "GeneraTokenDataCreditoPorOficina");
    }
    xmlElement = xmlDocument.DocumentElement;
    return xmlElement;
}

```

Figura 123 Configuraciones extras – Convertir a XmlElement 2

- Se actualiza la interfaz correspondiente

```

TokenWSI.cs x TokenWS.cs Usuario_OficialDivisionPoLWSI.cs Usuario_OficialDivisionPoLWSI.cs Web.config
WCFInterfaces.RepositorioToken.ITokenWS GeneraTokenDataCreditoPorOficina(SecuencialME mensajeEntrada)
    /// </summary>
    /// <param name="mensajeEntrada">Mensaje de entrada con el secuencial</param>
    /// <returns>Mensaje con el Token</returns>
    [SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
    [FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
    [OperationContract(Name = "GeneraTokenDataCreditosOficinasDesconectadas")]
    void GeneraTokenDataCreditosOficinasDesconectadas();

    [SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
    [FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndSign)]
    [OperationContract(Name = "GeneraTokenDataCreditoPorOficina")]
    //System.Xml.XmlDocument GeneraTokenDataCreditoPorOficina(SecuencialME mensajeEntrada);
    XmlElement GeneraTokenDataCreditoPorOficina(SecuencialME mensajeEntrada);

```

Figura 124 Configuraciones extras - Actualizar interfaz 2

- Convertir XmlDocument a XElement en FBSServicios/RepositorioToken/TokenWS.cs

```

TokenWSI.cs TokenWS.cs x Usuario_OficialDivisionPoLWSI.cs Usuario_OficialDivisionPoLWSI.cs
FBSServicios.RepositorioToken.ITokenWS D:\FBS_WCF_4590\FBSServidor\FBSServicios\RepositorioToken\TokenWS.cs
public XmlDocument GeneraTokenPagoEnLote(PagoEnLoteTokenOficinaME mensajeEntrada)
{
    XmlDocument xmlDocument = new XmlDocument();

    try
    {
        List<Oficina> listaOficinasDesconectadas = OficinaActor.DameListaPorSecuencial();
        List<AbonoPrestamoEnLoteNoProcesadoOficinaME> listaAbonoPrestamoEnLoteComprometido = OficinaActor.DameListaPorSecuencial();

        foreach (Oficina oficinaDesconectada in listaOficinasDesconectadas)
        {
            List<AbonoPrestamoEnLoteNoProcesadoOficinaME> listaAbonoPrestamoEnLoteComprometido = OficinaActor.DameListaPorSecuencial();
            List<int> listaSecuencialesToken = new List<int>();
            List<int> listaSecuencialesAbonoPrestamo = new List<int>();

```

Figura 125 Configuraciones extras - Actualizar XmlDocument 3

- Pasa a ser.

```

public XmlElement GeneraTokenPagoEnLote(PagoEnLoteTokenOficinaME mensajeEntrada)
{
    XmlElement xmlElement;
    try
    {
        XmlDocument xmlDocument = new XmlDocument();
        List<Oficina> listaOficinasDesconectadas = OficinaActor.DameListaPorSecuencialEmpresaEstaActiva(this.settings.Empresa, t

        List<AbonoPrestamoEnLoteNoProcesadoOficinaME> listaAbonoPrestamoEnLoteCompleta = new List<AbonoPrestamoEnLoteNoProcesado

        xmlElement = xmlDocument.DocumentElement;
    }
    catch (Exception e)
    {
        throw RaiseException.Raise(e, "GeneraTokenDataCreditoPorOficina");
    }
    return xmlElement;
}

```

Figura 126 Configuraciones extras - Modificar XmlElement 3

- Se actualiza la interfaz correspondiente

```

/// </summary>
/// <param name="mensajeEntrada"></param>
/// <returns></returns>
[SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
[FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndS
[OperationContract(Name = "GuardaParametrosMonitor")]
void GuardaParametrosMonitor(ParametroMonitor mensajeEntrada);

[SoapHeader("settings", typeof(MessageSettings), Direction = SoapHeaderDirection.In)]
[FaultContractAttribute(typeof(MessageError), ProtectionLevel = ProtectionLevel.EncryptAndS
[OperationContract(Name = "GeneraTokenPagoEnLote")]
[System.Xml.XmlDocument] GeneraTokenPagoEnLote(PagoEnLoteTokenOficinaME mensajeEntrada);
XmlElement GeneraTokenPagoEnLote(PagoEnLoteTokenOficinaME mensajeEntrada);

```

Figura 127 Configuraciones extras - Actualizar interfaz 3

- Probar el ingreso al servicio TokenWS

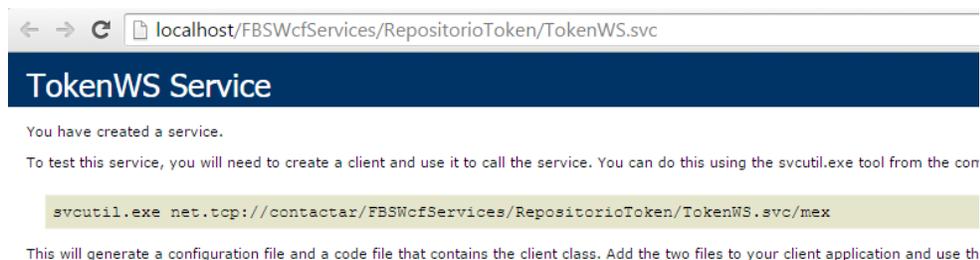


Figura 128 Configuraciones extras – Verificación de acceso 3

2.4.8. Configuraciones adicionales del equipo

2.4.8.1. Ampliar límite de tamaño del archivo web.config para IIS. Ingresar al Editor del Registro de Windows, escribiendo regedit en la aplicación Ejecutar.

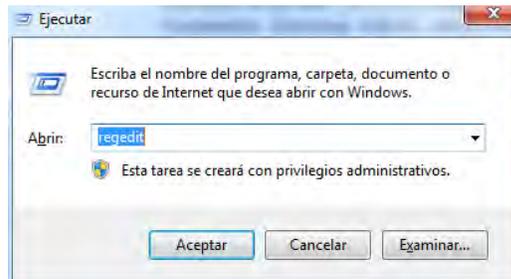


Figura 129 Regedit - Iniciar la funcionalidad editor del registro

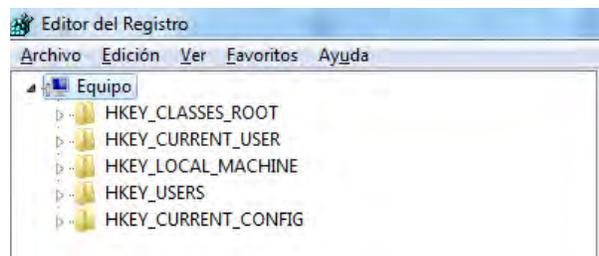


Figura 130 Regedit - Editor del registro iniciado

- Existen dos rutas en las cuales se van a hacer cambios y en ambas se debe realizar el mismo proceso

HKLM\Software\Wow6432Node\Microsoft\InetStp\

HKLM\Software\Microsoft\InetStp\Configuration\MaxWebConfigFileSizeInKB

- Se crea la clave *Configuration*

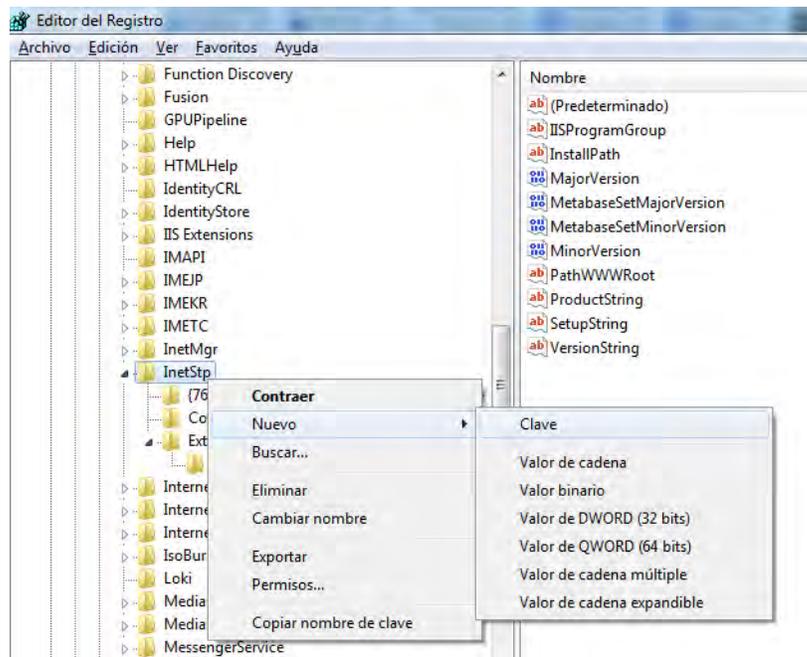


Figura 131 Regedit - Nueva clave

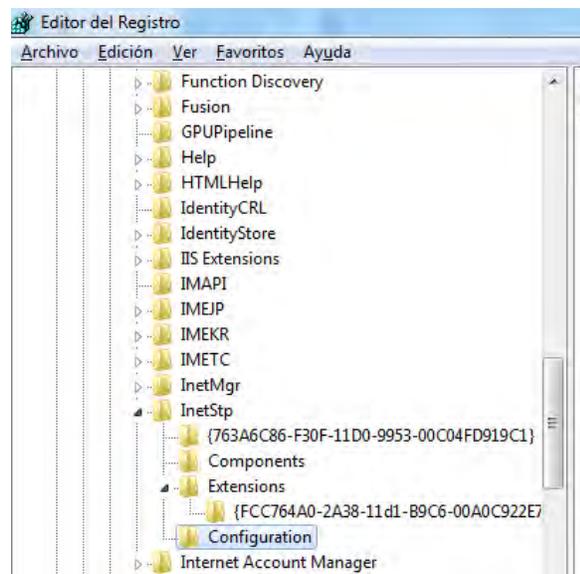


Figura 132 Regedit - Clave Configuration creada

- Se crea un nuevo valor de DWORD

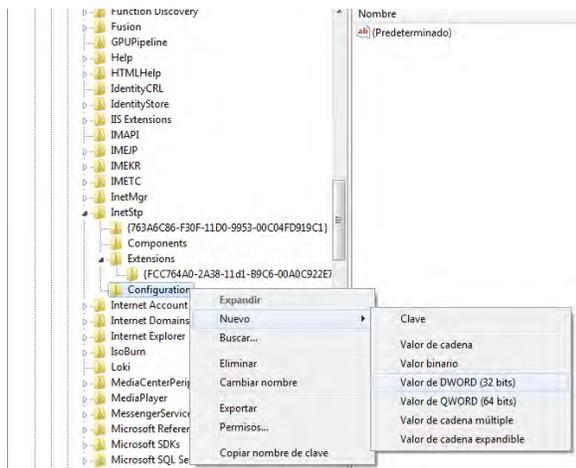


Figura 133 Regedit - Nuevo valor DWORD

- Ingresar el nombre *ConfigurationMaxWebConfigFileSizeInKB*

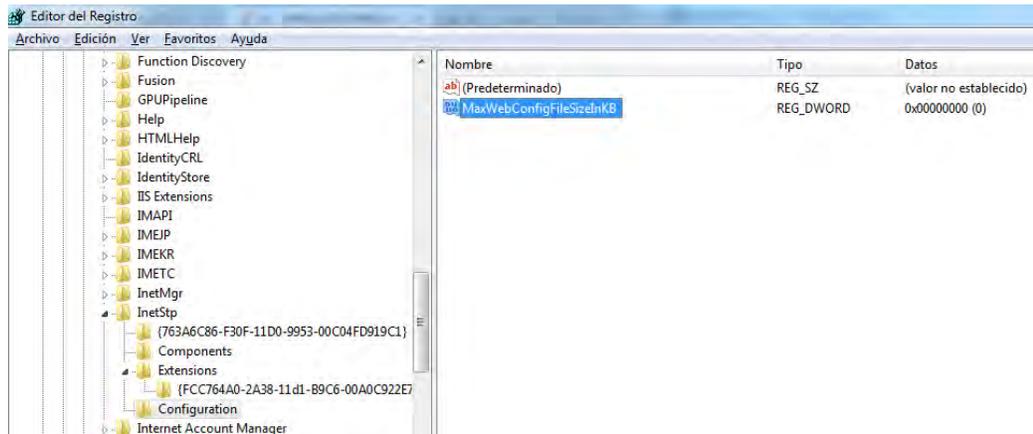


Figura 134 Regedit - Ingresar nombre de valor

- Se modifica el valor de la clave

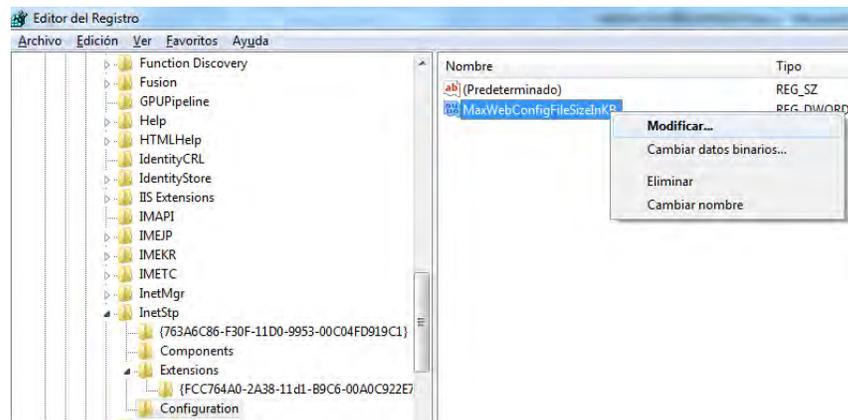


Figura 135 Regedit - Modificar valor de la clave

- Ingresar el nuevo límite de tamaño para el archivo web.config

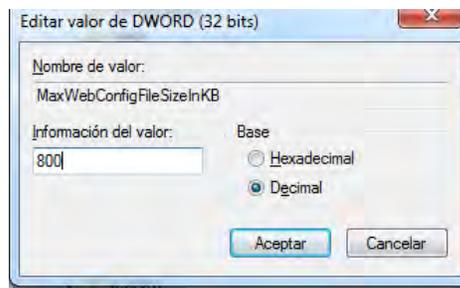


Figura 136 Regedit - Ingresar valor de la clave

Nombre	Tipo	Datos
(Predeterminado)	REG_SZ	(valor no establecido)
MaxWebConfigFileSizeInKB	REG_DWORD	0x00000320 (800)

Figura 137 Regedit - Valor creado

- Se reinicia el IIS, con el comando iisreset en la aplicación Ejecutar

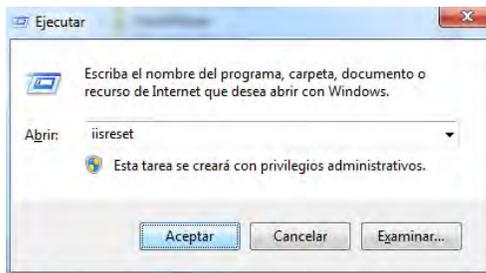


Figura 138 IIS - Reiniciar servicio del IIS

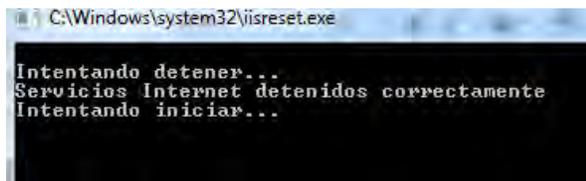


Figura 139 IIS - Reinicio de IIS en proceso

2.4.8.2. Dar permisos de acceso a la clave privada del certificado

- Ingresar al mmc

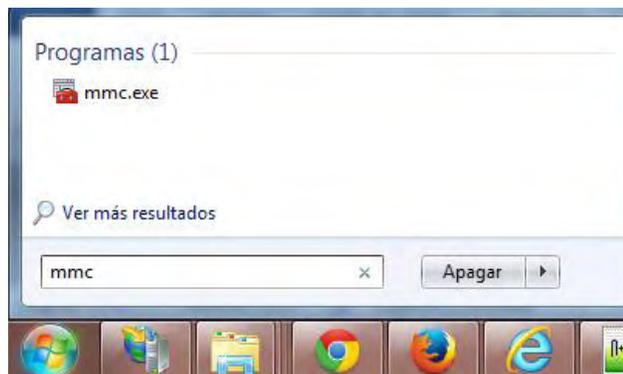


Figura 140 MMC - Iniciar la consola de administración de Microsoft

- Se verifica que el certificado WcfFinancal exista en Personal/Certificados

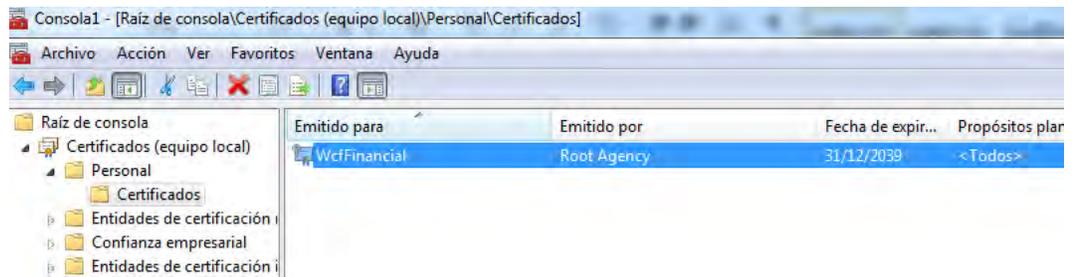


Figura 141 MMC - Verificar existencia del certificado WcfFinacial

- Seleccionar el certificado > click derecho > Administrar claves privadas...

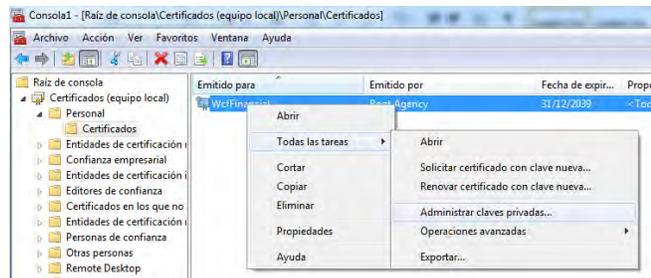


Figura 142 MMC - Administrar claves privadas del certificado

- Agregar con permisos de lectura al usuario que esté ejecutando la aplicación en el IIS

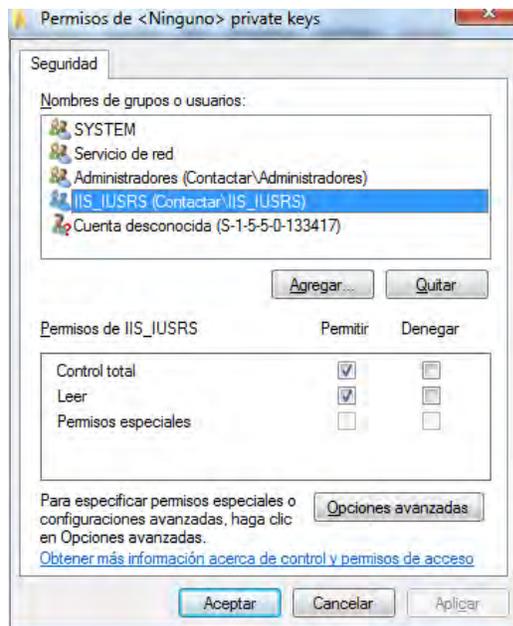


Figura 143 MMC - Asignar permiso de lectura a usuario de IIS

2.4.8.3. Habilitar servicios de net.tcp en Windows

- Ir a la funcionalidad de Windows con nombre *Activar o desactivar las características de Windows*.

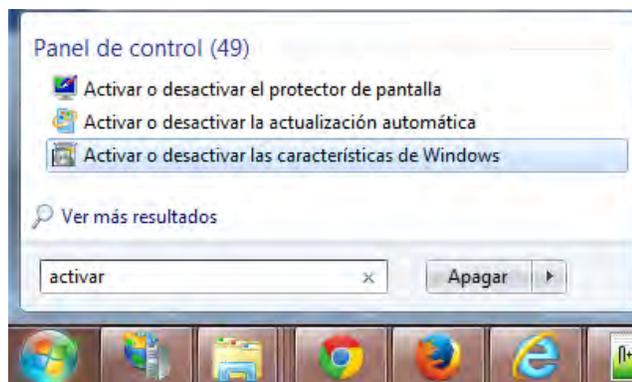


Figura 144 WCF - Iniciar la funcionalidad de características de Windows

- Activar las características de Windows Communication Foundation que se encuentran dentro de *Microsoft .NET Framework 3.5.1*

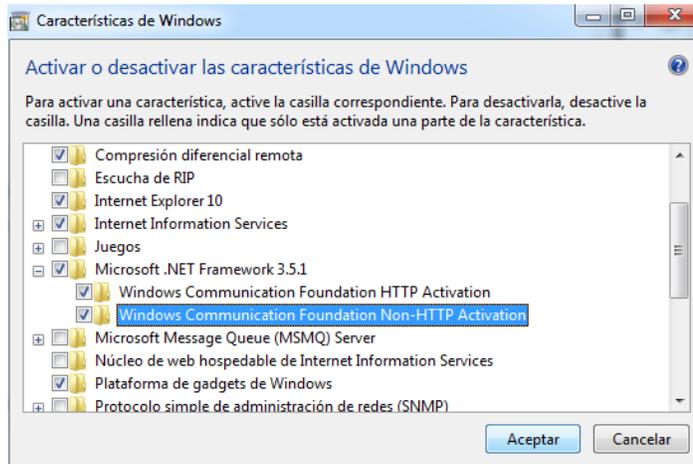


Figura 145 WCF - Activar características de WCF

- Abrir la funcionalidad *Servicios de Windows*

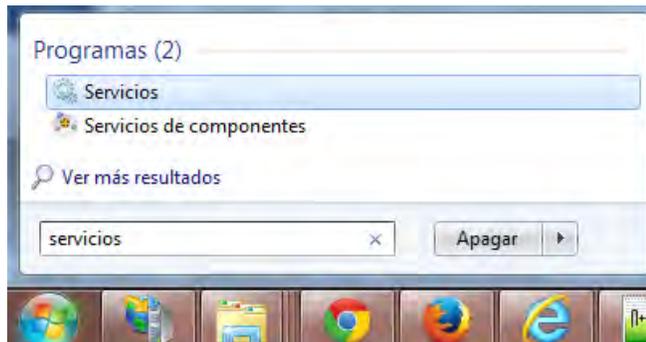


Figura 146 WCF - Iniciar la funcionalidad servicios

- Seleccionar los servicios *Net.Tcp Listener Adapter* y *Net.Tcp Port sharing Service* y posteriormente activar, configurar el inicio automático e iniciar cada uno de ellos.

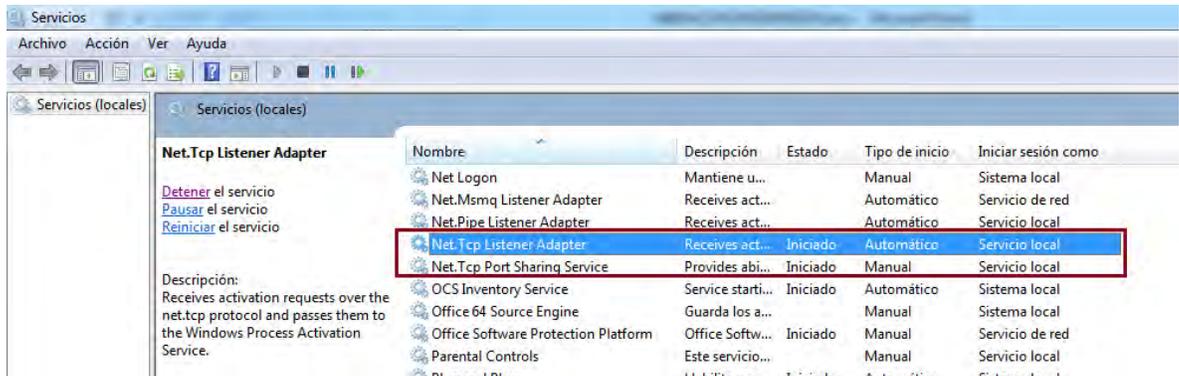


Figura 147 WCF - Configurar inicio de servicios de Net.Tcp

- Acceder al Command prompt de Visual Studio y ejecutar el comando [aspnet_regiis.exe -iru]

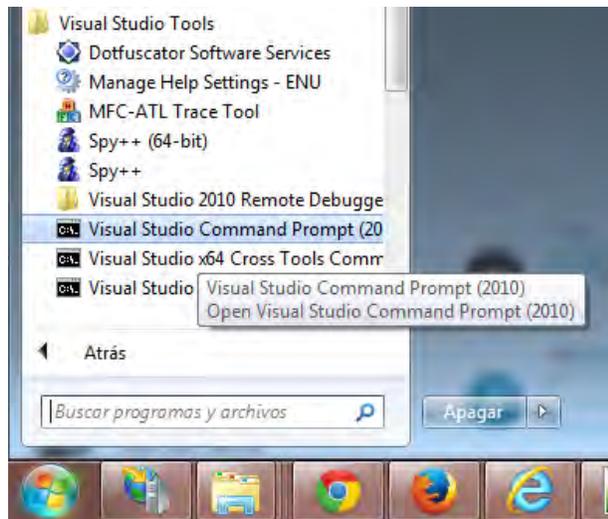


Figura 148 WCF - Iniciar consola de Visual Studio

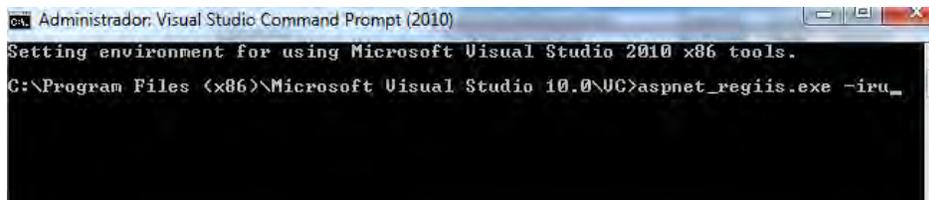


Figura 149 WCF - Ejecutar comando aspnet_regiis.exe -iru

- Por último reiniciar el IIS ingresando el comando *iisreset* en la funcionalidad *Ejecutar* de Windows

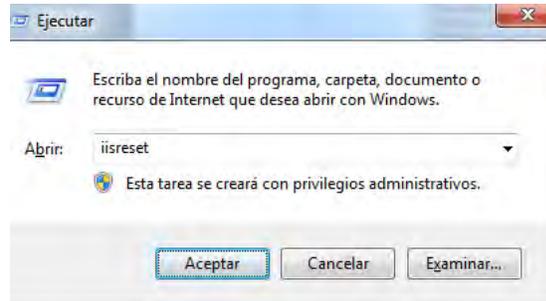


Figura 150 WCF - Reiniciar IIS

2.5. MIGRACIÓN SOLUCIÓN PROXIES

2.5.1. Cambiar a Framework 4.0 los proyectos del proxy

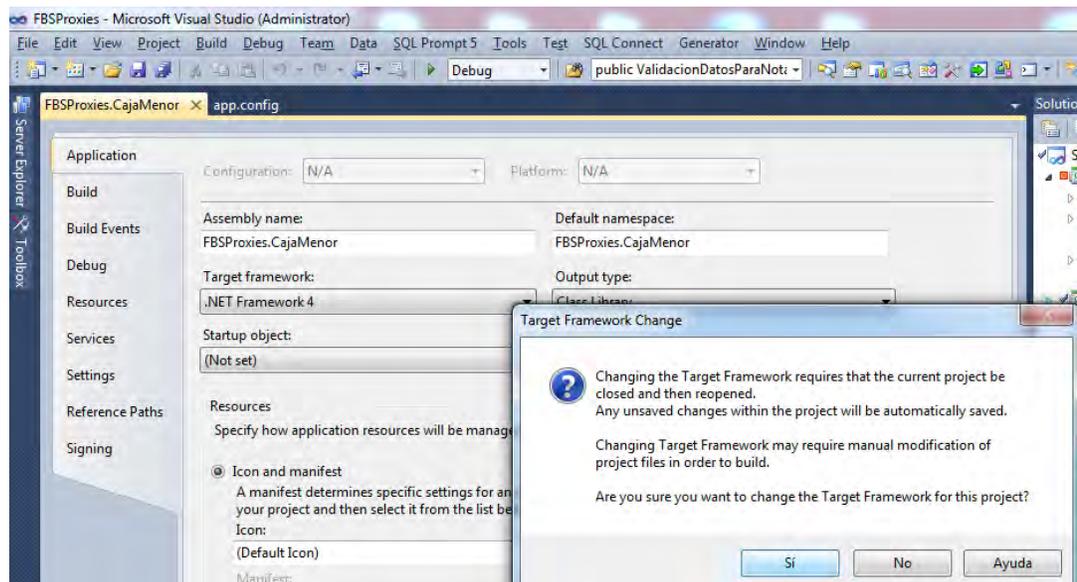


Figura 151 FBSProxies - Actualizar versión de .NET Framework

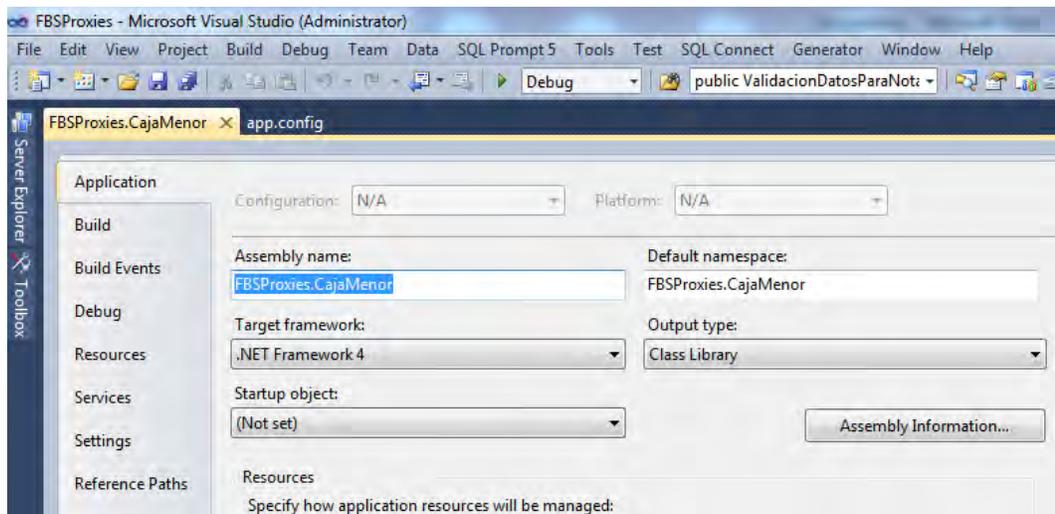


Figura 152 FBSPproxies - .NET Framework actualizado

2.5.2. Obtener links actualizados de los service references. *Buscar [http://localhost/] en el app.config de cada proyecto.*

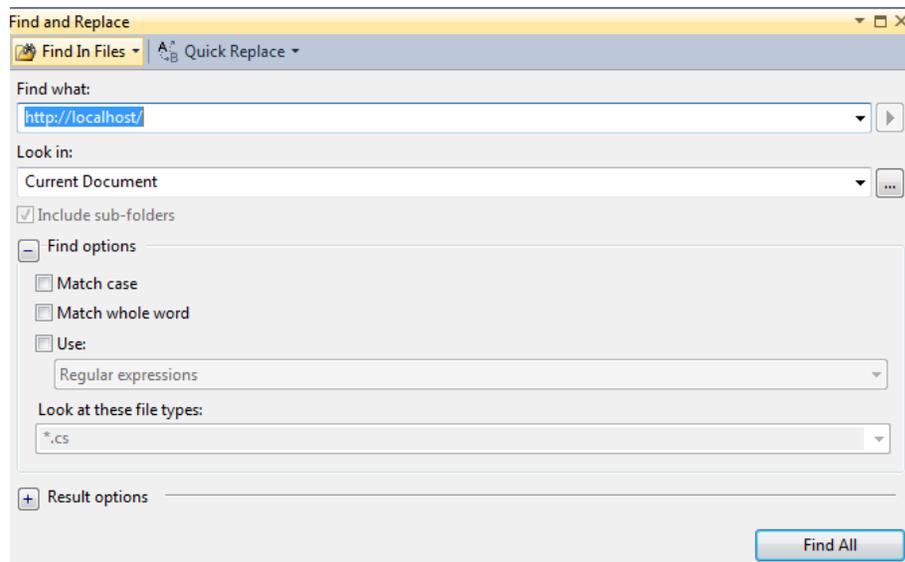


Figura 153 FBSPproxies - Buscar links de servicios

- Copiar el resultado a notepad++

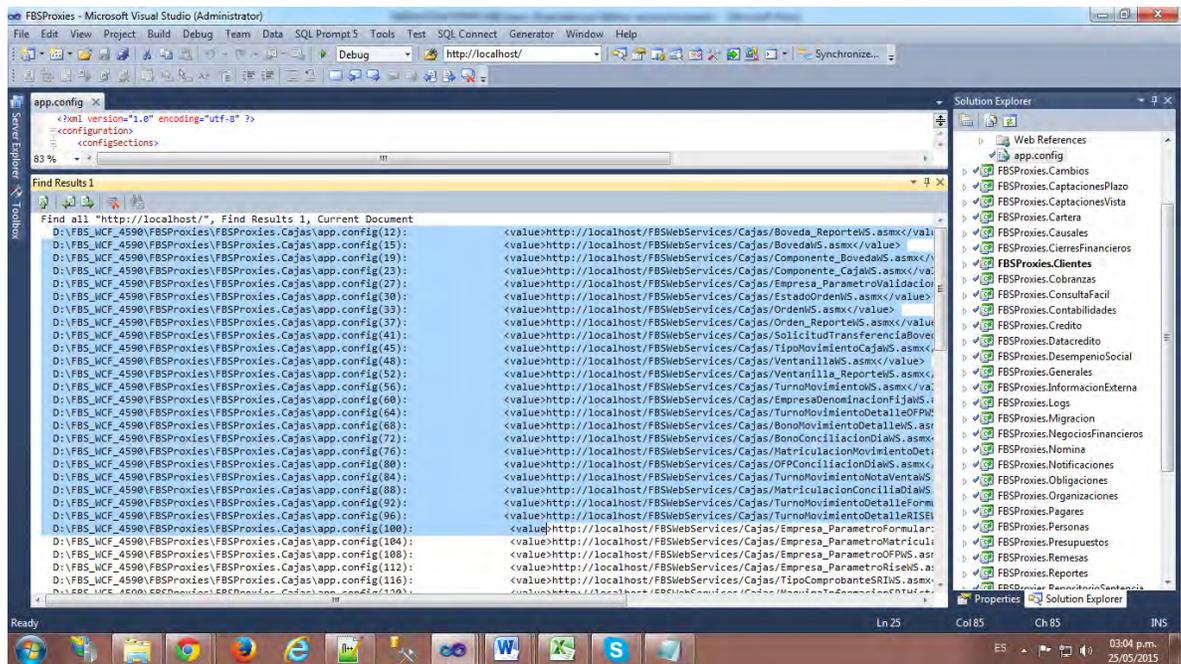


Figura 154 FBSProxies - Resultado de la búsqueda

- Eliminar las filas de acuerdo con la siguiente imagen para que las filas queden del mismo ancho inicial.

```

23 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(96): <value>ht
24 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(212): <value>h
25 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(100): <value>h
26 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(104): <value>h
27 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(108): <value>h
28 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(112): <value>h
29 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(116): <value>h
30 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(120): <value>h
31 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(124): <value>h
32 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(128): <value>h
33 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(132): <value>h
34 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(136): <value>h
35 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(140): <value>h
36 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(144): <value>h
37 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(148): <value>h
38 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(152): <value>h
39 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(156): <value>h
40 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(160): <value>h
41 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(164): <value>h
42 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(168): <value>h
43 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(172): <value>h
44 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(176): <value>h
45 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(180): <value>h
46 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(184): <value>h
47 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(188): <value>h
48 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(192): <value>h
49 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(196): <value>h
50 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(200): <value>h
51 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(204): <value>h
52 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(208): <value>h

```

Figura 155 FBSProxies - Ajustar extensión de filas

- Eliminar la parte inicial de cada fila de acuerdo con la siguiente imagen

```

17 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(72) : <value>http://localhost/FBSWebServices/Cajas/BonoCon
18 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(76) : <value>http://localhost/FBSWebServices/Cajas/Matricu
19 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(80) : <value>http://localhost/FBSWebServices/Cajas/OFPConc
20 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(84) : <value>http://localhost/FBSWebServices/Cajas/TurnoMo
21 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(88) : <value>http://localhost/FBSWebServices/Cajas/Matricu
22 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(92) : <value>http://localhost/FBSWebServices/Cajas/TurnoMo
23 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(96) : <value>http://localhost/FBSWebServices/Cajas/TurnoMo
24 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(21) : <value>http://localhost/FBSWebServices/Cajas/Usuario
25 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(10) : <value>http://localhost/FBSWebServices/Cajas/Empresa
26 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(10) : <value>http://localhost/FBSWebServices/Cajas/Empresa
27 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(10) : <value>http://localhost/FBSWebServices/Cajas/Empresa
28 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(11) : <value>http://localhost/FBSWebServices/Cajas/Empresa
29 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(11) : <value>http://localhost/FBSWebServices/Cajas/TipoCom
30 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(12) : <value>http://localhost/FBSWebServices/Cajas/Maquina
31 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(12) : <value>http://localhost/FBSWebServices/Cajas/Maquina
32 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(12) : <value>http://localhost/FBSWebServices/Cajas/Comprob
33 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(13) : <value>http://localhost/FBSWebServices/Cajas/Institu
34 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(13) : <value>http://localhost/FBSWebServices/Cajas/CodigoI
35 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(14) : <value>http://localhost/FBSWebServices/Cajas/Empresa
36 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(14) : <value>http://localhost/FBSWebServices/Cajas/Empresa
37 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(14) : <value>http://localhost/FBSWebServices/Cajas/Empresa
38 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(15) : <value>http://localhost/FBSWebServices/Cajas/Empresa
39 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(15) : <value>http://localhost/FBSWebServices/Cajas/Empresa
40 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(16) : <value>http://localhost/FBSWebServices/Cajas/Empresa
41 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(16) : <value>http://localhost/FBSWebServices/Cajas/Empresa
42 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(16) : <value>http://localhost/FBSWebServices/Cajas/Empresa
43 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(17) : <value>http://localhost/FBSWebServices/Cajas/Empresa
44 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(17) : <value>http://localhost/FBSWebServices/Cajas/CodigoI
45 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(18) : <value>http://localhost/FBSWebServices/Cajas/TurnoMo
46 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(18) : <value>http://localhost/FBSWebServices/Cajas/TurnoMo
47 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(18) : <value>http://localhost/FBSWebServices/Cajas/Empresa
48 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(19) : <value>http://localhost/FBSWebServices/Cajas/Oficina
49 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(19) : <value>http://localhost/FBSWebServices/Cajas/Empresa
50 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(20) : <value>http://localhost/FBSWebServices/Cajas/Impreso
51 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(20) : <value>http://localhost/FBSWebServices/Cajas/Oficina
52 D:\FBS_WCF_4590\FBSProxies\FBSProxies.Cajas\app.config(20) : <value>http://localhost/FBSWebServices/Cajas/Usuario

```

Figura 156 FBSProxies - Eliminar sección inicial de cada fila

- Pegar en el inicio de cada fila el texto net.tcp://localhost/FBSWcfServices

```

1 net.tcp://localhost/FBSWcfServices/ActivosFijos/CalculoWS.asmx</value>
2 net.tcp://localhost/FBSWcfServices/ActivosFijos/EstadoActivoWS.asmx</value>
3 net.tcp://localhost/FBSWcfServices/ActivosFijos/IndiceCorreccionWS.asmx</value>
4 net.tcp://localhost/FBSWcfServices/ActivosFijos/ItemWS.asmx</value>
5 net.tcp://localhost/FBSWcfServices/ActivosFijos/Persona_ResponsableWS.asmx</value>
6 net.tcp://localhost/FBSWcfServices/ActivosFijos/RubroWS.asmx</value>
7 net.tcp://localhost/FBSWcfServices/ActivosFijos/SubItemWS.asmx</value>
8 net.tcp://localhost/FBSWcfServices/ActivosFijos/TrasladoWS.asmx</value>
9 net.tcp://localhost/FBSWcfServices/ActivosFijos/Empresa_ParametroEjecucionActivosFijosWS.asmx</value>
10 net.tcp://localhost/FBSWcfServices/ActivosFijos/DivisionEstructuraActivoFijo_CondicionWS.asmx</value>
11 net.tcp://localhost/FBSWcfServices/ActivosFijos/ActivoWS.asmx</value>
12 net.tcp://localhost/FBSWcfServices/ActivosFijos/AccesoriosWS.asmx</value>
13 net.tcp://localhost/FBSWcfServices/ActivosFijos/Activo_BajaWS.asmx</value>
14 net.tcp://localhost/FBSWcfServices/ActivosFijos/ActivosFijosReportesWS.asmx</value>
15 net.tcp://localhost/FBSWcfServices/ActivosFijos/TrasladoNoDepreciadosADepreciadosWS.asmx</value>
16 net.tcp://localhost/FBSWcfServices/ActivosFijos/ActivoOficinaProrratoWS.asmx</value>
17 net.tcp://localhost/FBSWcfServices/ActivosFijos/MotivoActivoBajaWS.asmx</value>
18 net.tcp://localhost/FBSWcfServices/ActivosFijos/Activo_PersonalizadoWS.asmx</value>
19 net.tcp://localhost/FBSWcfServices/ActivosFijos/TrasladoDetalleWS.asmx</value>
20 net.tcp://localhost/FBSWcfServices/ActivosFijos/ActivoComprobanteCtablePaqoWS.asmx</value>
21 net.tcp://localhost/FBSWcfServices/ActivosFijos/Activo_FacturaCotizacionWS.asmx</value>
22 net.tcp://localhost/FBSWcfServices/ActivosFijos/MejorasWS.asmx</value>

```

Figura 157 FBSProxies - Insertar url de conexión

- Buscar y reemplazar

Buscar: [.asmx</value>]

Reemplazar por: [.svc/mex]

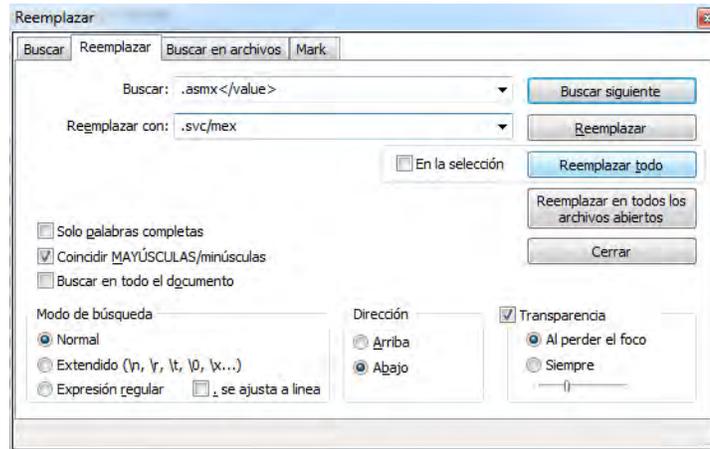


Figura 158 FBSProxies - Reemplazar extensión de los links

- Se tienen los links actualizados para las service references

```

1 net.tcp://localhost/WcfServicioIIS/Cajas/Boveda_ReporteWS.svc/mex
2 net.tcp://localhost/WcfServicioIIS/Cajas/BovedaWS.svc/mex
3 net.tcp://localhost/WcfServicioIIS/Cajas/Componente_BovedaWS.svc/mex
4 net.tcp://localhost/WcfServicioIIS/Cajas/Componente_CajaWS.svc/mex
5 net.tcp://localhost/WcfServicioIIS/Cajas/Empresa_ParametroValidacionCajasWS.svc/mex
6 net.tcp://localhost/WcfServicioIIS/Cajas/EstadoOrdenWS.svc/mex
7 net.tcp://localhost/WcfServicioIIS/Cajas/OrdenWS.svc/mex
8 net.tcp://localhost/WcfServicioIIS/Cajas/Orden_ReporteWS.svc/mex
9 net.tcp://localhost/WcfServicioIIS/Cajas/SolicitudTransferenciaBovedaWS.svc/mex
10 net.tcp://localhost/WcfServicioIIS/Cajas/TipoMovimientoCajaWS.svc/mex
11 net.tcp://localhost/WcfServicioIIS/Cajas/VentanillaWS.svc/mex
12 net.tcp://localhost/WcfServicioIIS/Cajas/Ventanilla_ReporteWS.svc/mex
13 net.tcp://localhost/WcfServicioIIS/Cajas/TurnoMovimientoWS.svc/mex
14 net.tcp://localhost/WcfServicioIIS/Cajas/EmpresaDenominacionFijaWS.svc/mex
15 net.tcp://localhost/WcfServicioIIS/Cajas/TurnoMovimientoDetalleOFFWS.svc/mex
16 net.tcp://localhost/WcfServicioIIS/Cajas/BonoMovimientoDetalleWS.svc/mex
17 net.tcp://localhost/WcfServicioIIS/Cajas/BonoConciliacionDiaWS.svc/mex
18 net.tcp://localhost/WcfServicioIIS/Cajas/MatriculacionMovimientoDetalleWS.svc/mex
19 net.tcp://localhost/WcfServicioIIS/Cajas/OFPConciliacionDiaWS.svc/mex
20 net.tcp://localhost/WcfServicioIIS/Cajas/TurnoMovimientoNotaVentaWS.svc/mex
21 net.tcp://localhost/WcfServicioIIS/Cajas/MatriculacionConciliaDiaWS.svc/mex
22 net.tcp://localhost/WcfServicioIIS/Cajas/TurnoMovimientoDetalleFormularioSRIWS.svc/mex
23 net.tcp://localhost/WcfServicioIIS/Cajas/TurnoMovimientoDetalleRISEWS.svc/mex
24 net.tcp://localhost/WcfServicioIIS/Cajas/UsuarioAutorizaResponsableCierreCajaWS.svc/mex
25 net.tcp://localhost/WcfServicioIIS/Cajas/Empresa_ParametroFormularioSRIWS.svc/mex
26 net.tcp://localhost/WcfServicioIIS/Cajas/Empresa_ParametroMatriculacionWS.svc/mex
27 net.tcp://localhost/WcfServicioIIS/Cajas/Empresa_ParametroOFFWS.svc/mex
28 net.tcp://localhost/WcfServicioIIS/Cajas/Empresa_ParametroRiseWS.svc/mex
29 net.tcp://localhost/WcfServicioIIS/Cajas/TipoComprobanteSRIWS.svc/mex
30 net.tcp://localhost/WcfServicioIIS/Cajas/MaquinaInformacionSRIHistoricoWS.svc/mex

```

Figura 159 FBSProxies - Listado de links para las service references

2.5.3. Importar y configurar librería WCFExtras. Agregar el código XML para importar la librería WCFExtras.

Buscar: [\<\client\>]

Reemplazar por: [<metadata>\n\t\t\t<wsdlImporters>\n\t\t\t\t<extension type="WCFExtras.Soap.SoapHeaderImporter, WCFExtras" />\n\t\t\t</wsdlImporters>\n\t\t</metadata>\n\t</client>]

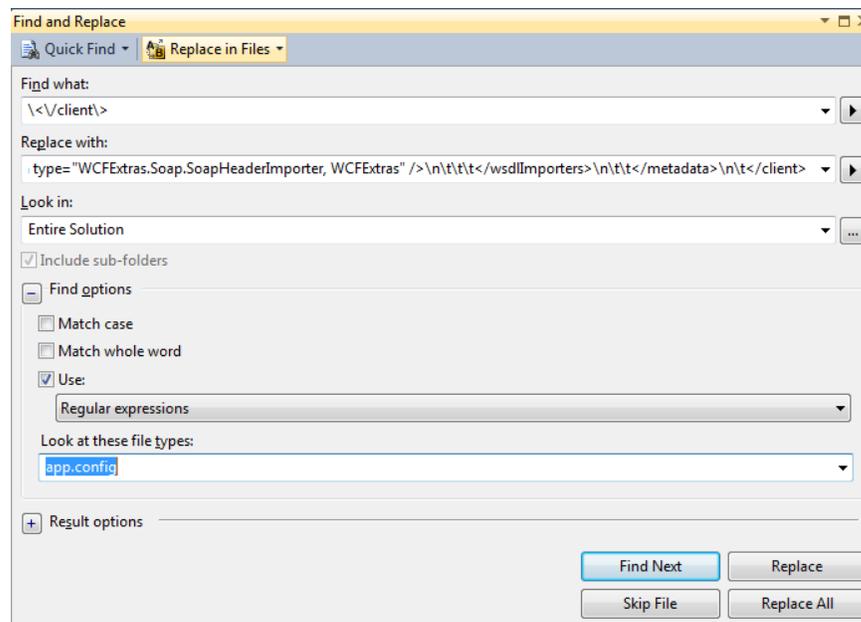


Figura 160 FBSPproxies - Insertar código XML para usar librería WCFExtras

- Guardar Todo
- Agregar referencia a WCFExtras

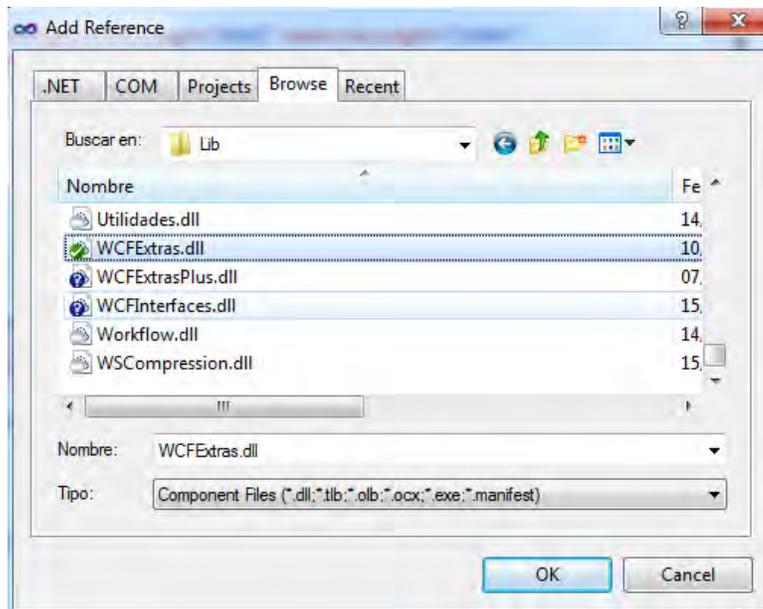


Figura 161 FBSPproxies - Agregar DLL de WCFExtras

- Incluir en el app.config de cada proyecto, en la sección client de la etiqueta system.serviceModel la siguiente configuración:

```
<metadata>
```

```
<wsdlImporters>
```

```
<extension type="WCFExtras.Soop.SoopHeaderImporter, WCFExtras" />
```

```
</wsdlImporters>
```

```
</metadata>
```

```
</endpoint>
<endpoint address="net.tcp://172.16.18.8/FBSWcfServices/ActivosFijos/Activo_BajaWS.svc"
  binding="netTcpBinding" bindingConfiguration="NetTcpBindingEndpoint1"
  contract="Activo_Baja.IActivo_BajaWS" name="NetTcpBindingEndpoint1">
  <identity>
    <dns value="WcfFinanciam" />
  </identity>
</endpoint>
<metadata>
  <wsdlImporters>
    <extension type="WCFExtras.Soap.SoapHeaderImporter, WCFExtras" />
  </wsdlImporters>
</metadata>
</client>
</system.serviceModel>
</configuration>
```

Figura 162 FBSProxies - Ejemplo etiqueta metadata para el app.config

- Agregar referencia a WCFExtras.dll en todos los proyectos.
- Eliminar las carpetas que se encuentren excluidas del proyecto.

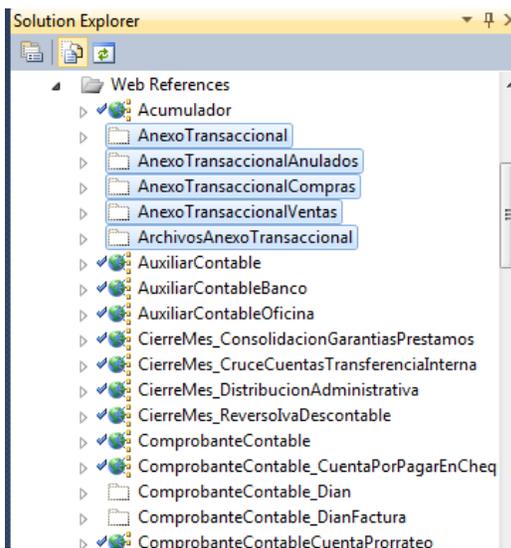


Figura 163 FBSProxies - Eliminar carpetas excluidas del proyecto

2.5.4. Ejecutar macro de actualización de service references. Abrir el macro para actualizar las service references: FBSProxies_ActualizarWebReferences, ubicarse en el método RecorrerSolucion() y ejecutar el macro.

- Verificar que la constante ACTUALIZAR_TODAS tenga asignado el valor True.

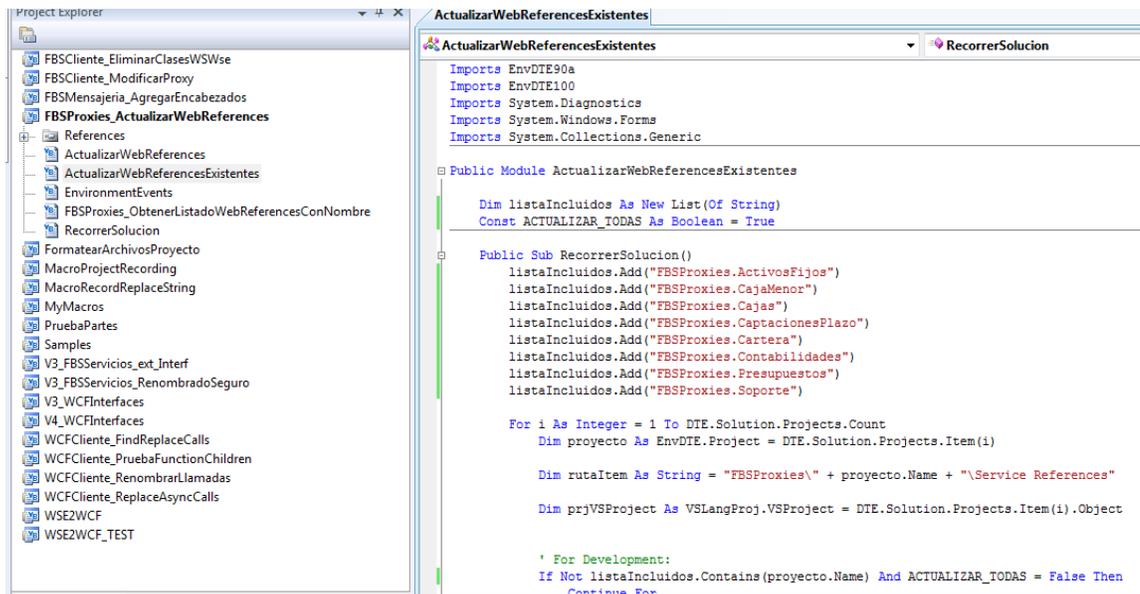


Figura 164 FBSProxies - Macro FBSProxies_ActualizarWebReferences

2.6. MIGRACIÓN SOLUCIÓN CLIENTE

2.6.1. Actualizar todos los proyectos a Framework 4.0. Pasar todos los proyectos de la solución FBSCliente a Framework 4.0.

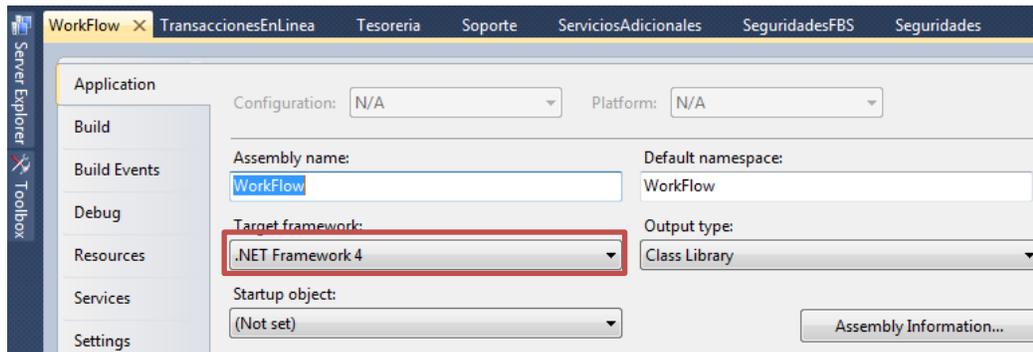


Figura 165 FBSCliente - Actualizar proyectos a .NET Framework 4.0

2.6.2. Ejecutar macro de actualización de proyecto FBSProxies. Abrir algún archivo de FBSProxies, abrir el Macro FBSCliente_ModificarProxy, ubicarse en ProcesarProyecto y ejecutar (F5).

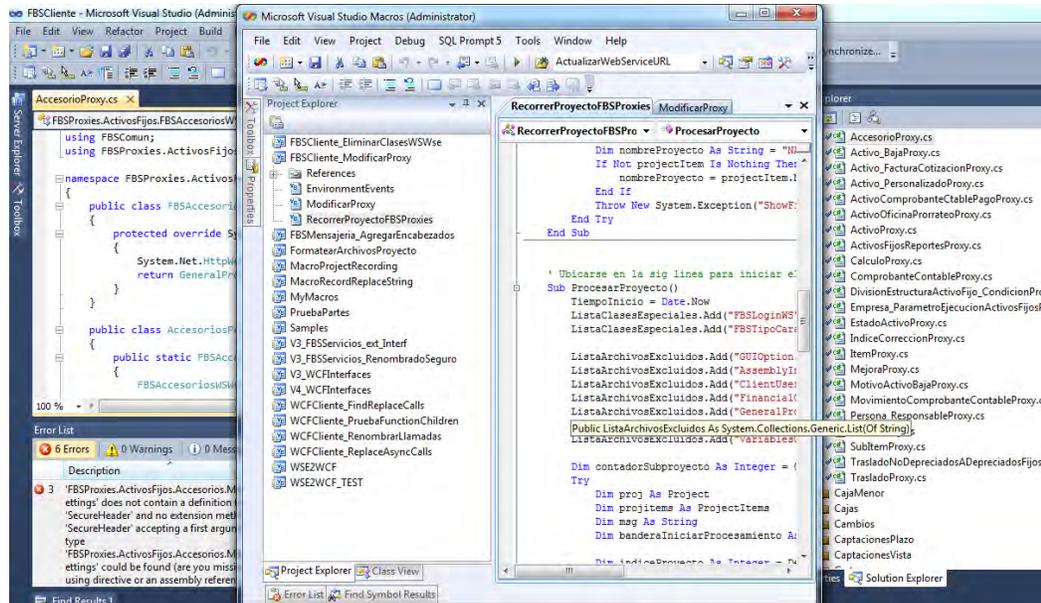


Figura 166 FBSCliente - Macro FBSCliente_ModificarProxy

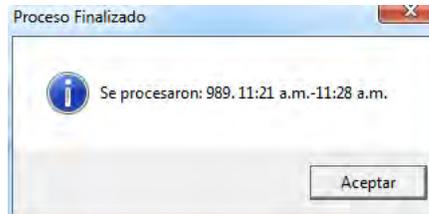


Figura 167 FBSCliente - Datos de macro FBSCliente_ModificarProxy

- Agregar Carpeta WcfComun a FBSProxies

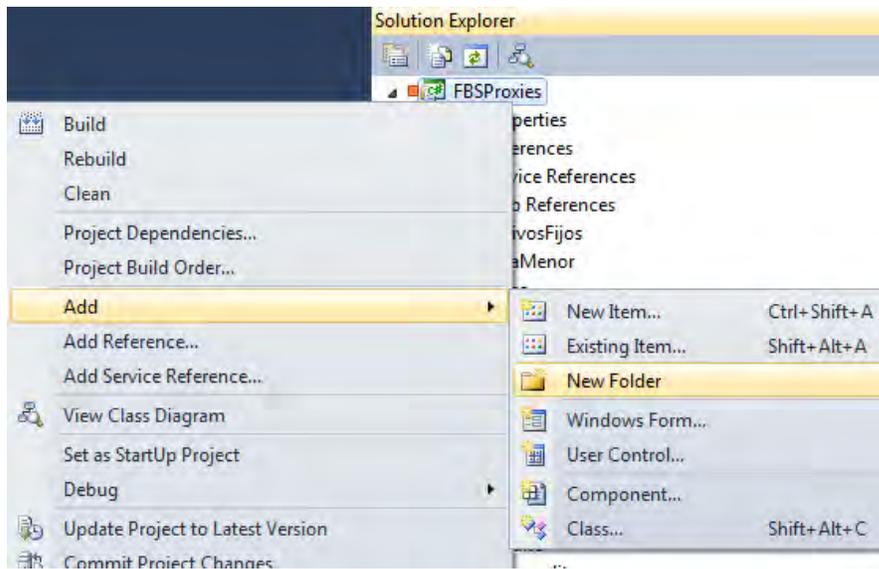


Figura 168 FBSCliente - Agregar carpeta WcfComun



Figura 169 FBSCliente - Carpeta WcfComun creada

- Copiar las clases (3) que se encuentran en Recursos/FBSCliente en WcfComun

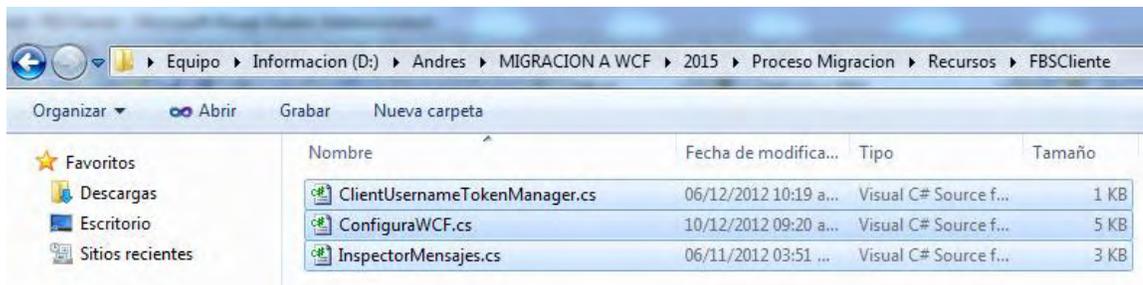


Figura 170 FBSCliente - Importar clases a WcfComun

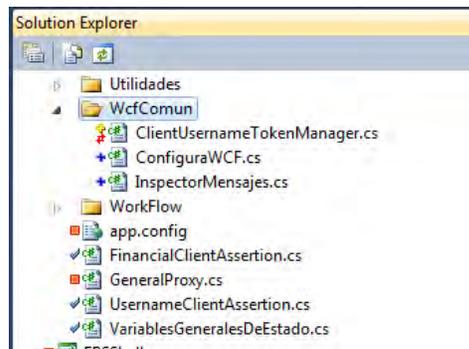


Figura 171 FBSCliente - Clases importadas a WcfComun

2.6.3. Referencias. En todos los proyectos, agregar referencia a las siguientes librerías:

- WCFExtras
- System.ServiceModel
- System.Runtime.Serialization

Se deben agregar los siguientes using:

- using System.ServiceModel;
- using System.Security.Cryptography.X509Certificates;
- using System.ServiceModel.Security;
- using WCFExtras.Soop;

Buscar: [namespace]

Reemplazar `por:` `[using System.ServiceModel;\nusing System.Security.Cryptography.X509Certificates;\nusing System.ServiceModel.Security;\nusing WCFExtras.Soop;\nnamespace]`

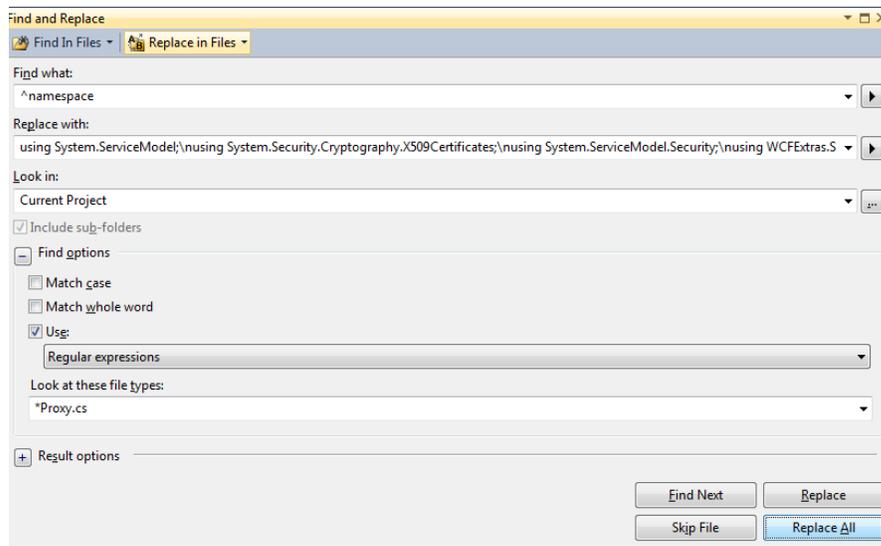


Figura 172 FBSCliente - Agregar using para comunicación con WCF

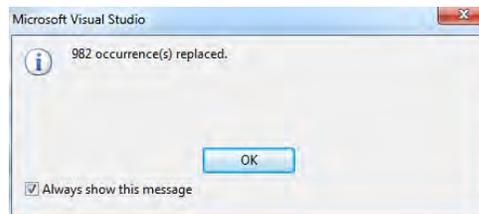


Figura 173 FBSCliente - Sentencias de nuevos using agregadas

2.6.4. Excluir archivos de la solución. Ubicarse en el proyecto FBSProxies y excluir los siguientes archivos:

- Datacredito/AlertaProxy.cs
- Contabilidades/ConceptoReteFuente_CuentaContableProxy.cs

Ubicarse en el proyecto Cartera y excluir el proyecto:

- Transaccion/DetalleCuentaTransferenciaPrestamoMaestro_Presentador.cs

2.6.5. Actualizar excepciones. Realizar el siguiente reemplazo:

Buscar: [catch (SoapException)]

Reemplazar por: [catch (FaultException)]

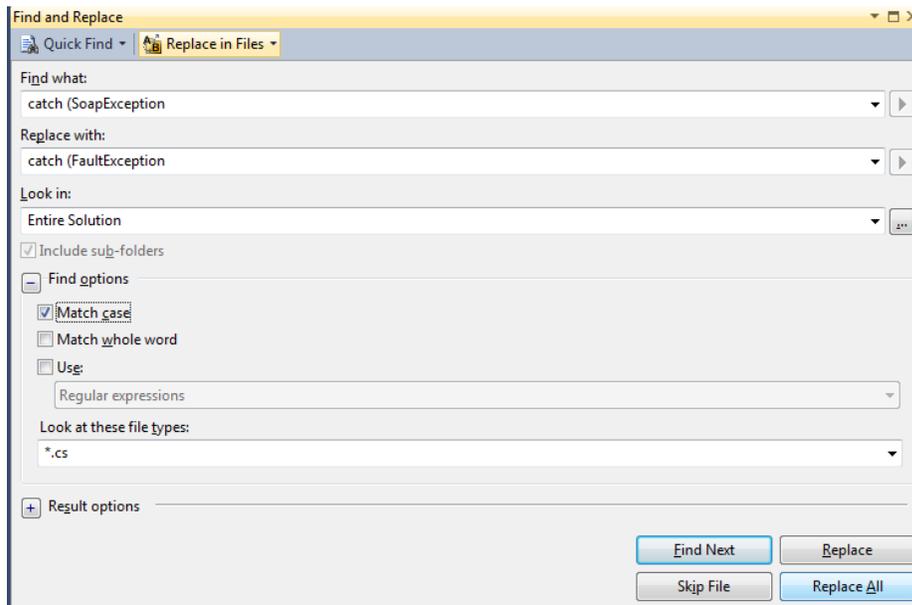


Figura 174 FBSCliente - Actualizar tipo de excepción

- En FBSComun.UtilitarioWindows.cs el código del método ProcesaSoapException debe quedar como se encuentra en la siguiente figura:

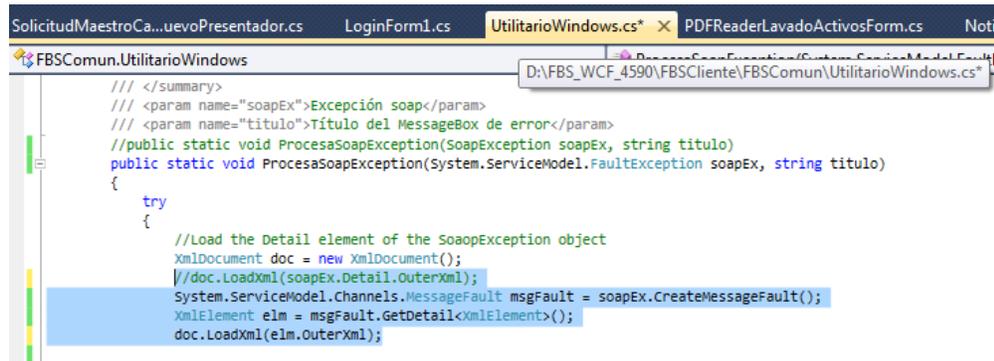


Figura 175 FBSCliente - Actualizar código de ProcesaSoapException

- Buscar: [if (ex.InnerException != null && ex.InnerException.GetType() == typeof(SoapException))]
- Reemplazar por: [if (ex.InnerException != null && ex.InnerException.GetType() == typeof(System.ServiceModel.FaultException))]

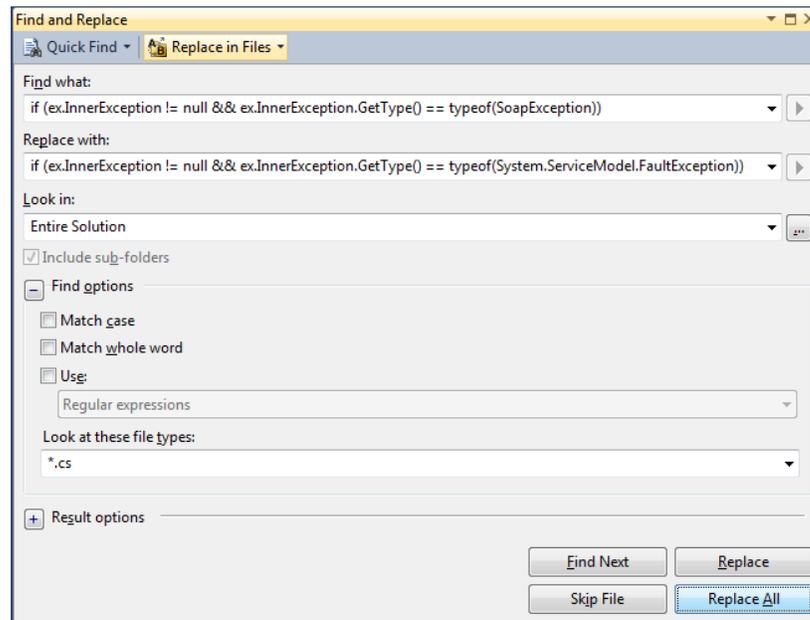


Figura 176 FBSCiente - Actualizar llamadas a excepciones

Buscar: [UtilitarioWindows.ProcesaSoapException((ex.InnerException as SoapException),]

Reempazar por: [UtilitarioWindows.ProcesaSoapException((ex.InnerException as System.ServiceModel.FaultException),]

Buscar: [== typeof(SoapException))]

Reemplazar por: [== typeof(System.ServiceModel.FaultException))]

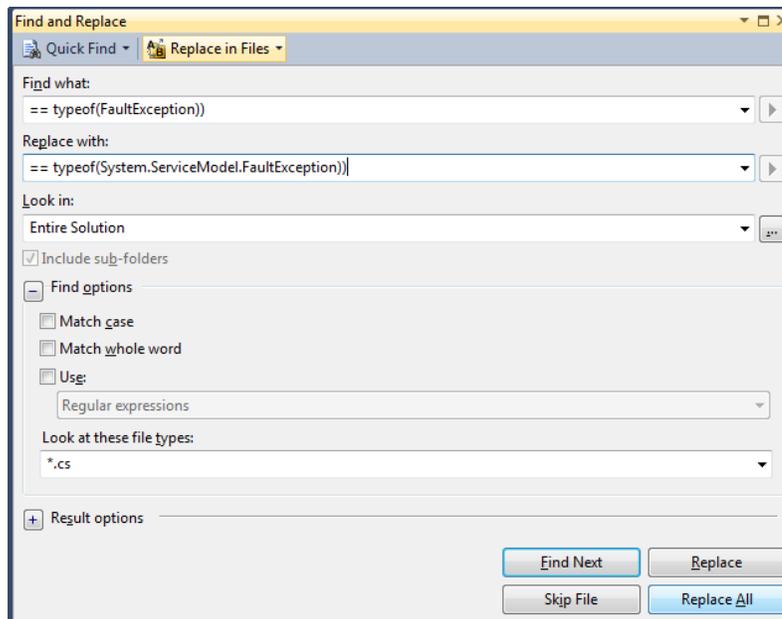


Figura 177 FBSCliente - Actualizar verificación de tipos de excepción

2.6.6. Buscar y reemplazar. Realizar el siguiente reemplazo:

Buscar: [using FBSProxies.ActivosFijos.Activo_Baja;]

Reemplazar por: [using FBSProxies.ActivosFijos.Activo_BajaWS;]

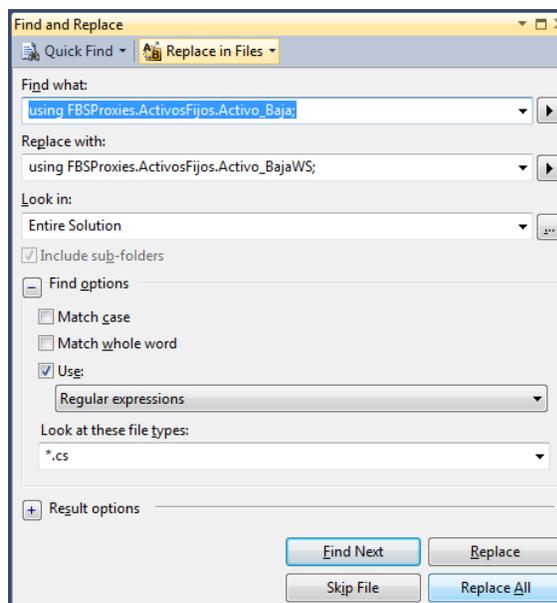


Figura 178 FBSCliente - Actualizar using de Activo_Baja

En el proyecto FBSProxies, buscar: `^[(\ \n)*]$`

Reemplazar por: `[]` (vacío)

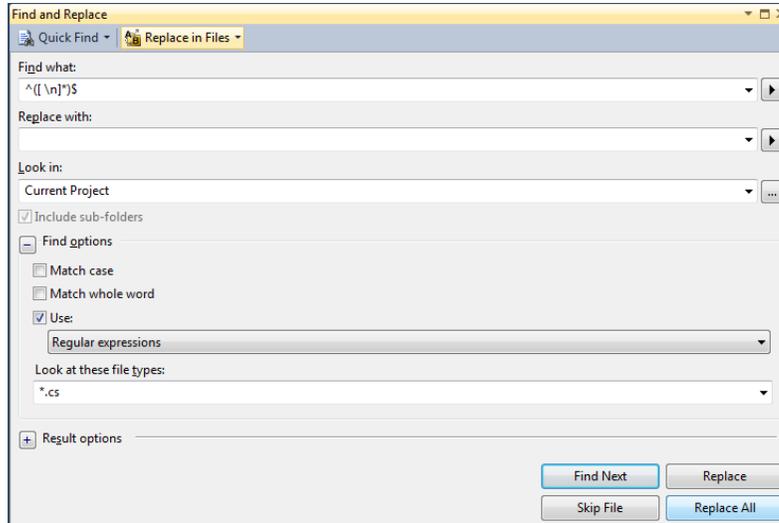


Figura 179 FBSCliente - Buscar y reemplazar

2.6.7. Ajustes finales. Agregar una setting de proyecto para indicar el nombre del certificado y el puerto del servicio.

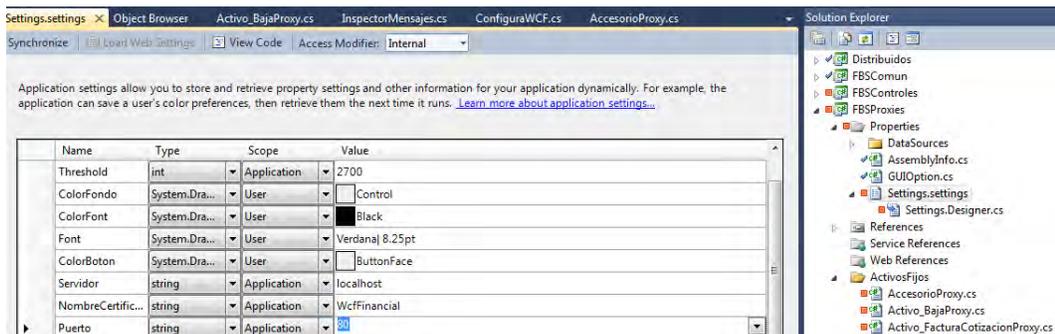


Figura 180 FBSCliente - Agregar nuevas settings

- Convertir XElement a XmlNode:

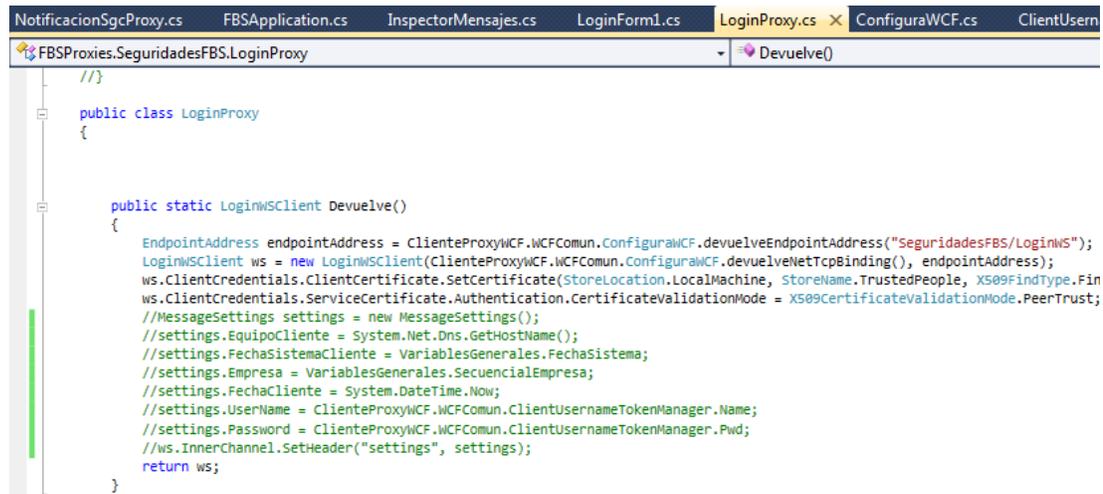
```
System.Xml.Linq.XElement xE = DevuelveXElement();
```

```
XmlDocument xD = new XmlDocument();
```

```
xD.LoadXml(xE.ToString());
```

```
XmlNode xmlNode = xD.FirstChild;
```

- Comentar en el archivo FBSProxies.SeguridadesFBS/LoginProxy.cs la sección de MessageSettings



```
NotificacionSgcProxy.cs  FBSApplication.cs  InspectorMensajes.cs  LoginForm1.cs  LoginProxy.cs  ConfiguraWCF.cs  ClientUsern
FBSProxies.SeguridadesFBS.LoginProxy  Devuelve()
//}
public class LoginProxy
{
    public static LoginWSClient Devuelve()
    {
        EndpointAddress endpointAddress = ClienteProxyWCF.WCFComun.ConfiguraWCF.devuelveEndpointAddress("SeguridadesFBS/LoginWS");
        LoginWSClient ws = new LoginWSClient(ClienteProxyWCF.WCFComun.ConfiguraWCF.devuelveNetTcpBinding(), endpointAddress);
        ws.ClientCredentials.ClientCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.TrustedPeople, X509FindType.Find
        ws.ClientCredentials.ServiceCertificate.Authentication.CertificateValidationMode = X509CertificateValidationMode.PeerTrust;
        //MessageSettings settings = new MessageSettings();
        //settings.EquipoCliente = System.Net.Dns.GetHostName();
        //settings.FechaSistemaCliente = VariablesGenerales.FechaSistema;
        //settings.Empresa = VariablesGenerales.SecuencialEmpresa;
        //settings.FechaCliente = System.DateTime.Now;
        //settings.UserName = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Name;
        //settings.Password = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Pwd;
        //ws.InnerChannel.SetHeader("settings", settings);
        return ws;
    }
}
```

Figura 181 FBSCiente - Actualizar código Devuelve() de LoginProxy.cs

- Agregar a UsuarioComplementoProxy el siguiente código

```
public static UsuarioComplementoWSClient Devuelve(string usuario, string
password)
{
    EndpointAddress endpointAddress =
    ClienteProxyWCF.WCFComun.ConfiguraWCF.devuelveEndpointAddress("Segurid
ades.UsuarioComplementoWS");
    UsuarioComplementoWSClient ws = new
    UsuarioComplementoWSClient(ClienteProxyWCF.WCFComun.ConfiguraWCF.dev
uelveNetTcpBinding(), endpointAddress);
}
```

```
ws.ClientCredentials.ClientCertificate.SetCertificate(StoreLocation.LocalMachine,  
StoreName.Root, X509FindType.FindBySubjectName,  
FBSProxies.Properties.Settings.Default.NombreCertificado);
```

```
MessageSettings settings = new MessageSettings();  
settings.EquipoCliente = System.Net.Dns.GetHostName();  
settings.FechaSistemaCliente = VariablesGenerales.FechaSistema;  
settings.Empresa = VariablesGenerales.SecuencialEmpresa;  
settings.FechaCliente = System.DateTime.Now;  
settings.UserName = System.Threading.Thread.CurrentPrincipal.Identity.Name;  
settings.Password = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Pwd;  
ws.InnerChannel.SetHeader("settings", settings);  
return ws;  
}
```

- Copiar este código a UsuarioComplementoProxy

```

UsuarioComplementoProxy.cs  Clave_EdicionPresentador.cs
FBSProxies.Seguridades.UsuarioComplementoProxy  Devuelve(string usuario, string password)
settings.Empresa = VariablesGenerales.SecuencialEmpresa;
settings.FechaCliente = System.DateTime.Now;
settings.UserName = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Name;
settings.Password = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Pwd;
ws.InnerChannel.SetHeader("settings", settings);
return ws;
}
}
public static UsuarioComplementoWSClient Devuelve(string usuario, string password)
{
    EndpointAddress endpointAddress = ClienteProxyWCF.WCFComun.ConfiguraWCF.devuelveEndpointAddress("Seguridades/Us
    UsuarioComplementoWSClient ws = new UsuarioComplementoWSClient(ClienteProxyWCF.WCFComun.ConfiguraWCF.devuelveEnd
    ws.ClientCredentials.ClientCertificate.SetCertificate(StoreLocation.LocalMachine, StoreName.Root, X509FindType.F
}
}
MessageSettings settings = new MessageSettings();
settings.EquipoCliente = System.Net.Dns.GetHostName();
settings.FechaSistemaCliente = VariablesGenerales.FechaSistema;
settings.Empresa = VariablesGenerales.SecuencialEmpresa;
settings.FechaCliente = System.DateTime.Now;
settings.UserName = System.Threading.Thread.CurrentPrincipal.Identity.Name;
settings.Password = ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Pwd;
ws.InnerChannel.SetHeader("settings", settings);
return ws;
}
}
}
}

```

Figura 182 FBSCliente - Actualizar UsuarioComplementoProxy

- En LoginForm1.cs agregar el código resaltado en la siguiente figura:

```

LoginForm1.cs*  FBSApplication.cs
FBSShell  AutenticarUsuario()
string unaClaveAcceso = FormsAuthentication.HashPasswordForStoringInConfigFile(unUsuario + unaClave, "SHA1");
FBSProxies.SeguridadesFBS.Login.AutenticarME autenticarME = new FBSProxies.SeguridadesFBS.Login.AutenticarME();
autenticarME.Maquina = System.Net.Dns.GetHostName();
List<string> direcciones = new List<string>();
System.Net.IPAddress[] direccionesIp = System.Net.Dns.GetHostAddresses(System.Net.Dns.GetHostName());
for (int i = 0; i < direccionesIp.Length; i++)
    direcciones.Add((direccionesIp[i] as System.Net.IPAddress).ToString());
autenticarME.Maquina = System.Net.Dns.GetHostName();
autenticarME.Direcciones = direcciones.ToArray();
autenticarME.NumeroDeIntento = this.numeroDeIntentos++;
autenticarME.Usuario = unUsuario;
autenticarME.Password = unaClaveAcceso;
autenticarME.UsaHuelladigital = this.usaLectorCheckBox.Checked;
autenticarME.Version = System.Deployment.Application.ApplicationDeployment.IsNetworkDeployed ? System.Deployment.Application.
autenticarME.EsVersionProgramacion = !System.Deployment.Application.ApplicationDeployment.IsNetworkDeployed;
FBSProxies.SeguridadesFBS.Login.AutenticarMS autenticarMS = new FBSProxies.SeguridadesFBS.Login.AutenticarMS();
System.Threading.Thread.SetData(System.Threading.Thread.GetNamedDataSlot("Pwd"), unaClaveAcceso);
ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Pwd = unaClaveAcceso;
ClienteProxyWCF.WCFComun.ClientUsernameTokenManager.Name = unUsuario;
autenticarMS = LoginProxy.Devuelve().Autenticar(autenticarME);
Distribuidos.BaseDatos.FijaCadenaConexion(autenticarMS.CadenaConexionBaseDatosLocal);
if (!Distribuidos.BaseDatos.TestConexion())
    throw new ApplicationException("Conexión a la base de datos local fallida");

```

Figura 183 FBSCliente - Actualizar LoginForm1.cs

2.6.8. Compilar y corregir errores en caso de ser necesario. Compilar la solución FBSCliente.

3. CONCLUSIONES

- El estudio de las características de WCF y de Financial Business System permitió definir las configuraciones que posteriormente se implementaron en el ambiente de producción del software, además se constituyó como la base para la reestructuración y adaptación del código fuente de todas las soluciones de Visual Studio que componen Financial.
- La actualización de tecnología de Financial Business System permitió ampliar la variedad de aplicaciones cliente que pueden consumir los servicios web que hospeda el servidor, además posibilita separar la lógica del negocio de la implementación de los servicios, es decir, no se necesitan realizar modificaciones significativas cada vez que se necesiten implementar cambios en el modo de transmisión de los datos, en el protocolo de seguridad o en la configuración de los servicios.
- El hacer uso del protocolo TCP para la transmisión de los datos entre cliente y servidor permite mejorar la velocidad de envío y recepción debido a que el tamaño de los mensajes es disminuido considerablemente.
- La seguridad que provee WCF, junto al uso de la autenticación por certificados X509 permite mantener la confidencialidad en la comunicación, se verifica la autenticidad del mensaje, se previene la suplantación del cliente y del servidor, además se observó que al capturar los mensajes transmitidos por la red, el contenido ya no se podía interpretar directamente, tal como se realizaba anteriormente con WSE.
- Las pruebas de velocidad realizadas a los servicios web bajo diferentes configuraciones de WCF, mostraron que la implementación de mecanismos de seguridad que brindan mayor seguridad, tales como los certificados X509, disminuyen la velocidad de respuesta debido a la ejecución de operaciones adicionales, como la confirmación de la identidad de los extremos y el cifrado de mensajes.
- Las funcionalidades programadas para realizar la reestructuración de código en lote, redujeron considerablemente el tiempo utilizado en la migración del ambiente de producción, y por ende permitieron disminuir el periodo de interrupción de las actividades normales del equipo de programación de la Corporación.

4. RECOMENDACIONES

- Mantener el constante apoyo por parte de las directivas, dependencias y procesos de la Corporación para promover la mejora continua en los sistemas de información, debido a la enorme importancia que representa para una entidad financiera la adecuada gestión de su información.
- Implementar constantemente actividades que promuevan la investigación de nuevas tendencias y tecnologías emergentes aplicables al contexto de la organización, que permitan mejorar el rendimiento, seguridad, confiabilidad, robustez y mantenibilidad de las aplicaciones desarrolladas y soportadas por el área de programación.
- Considerar la posibilidad de hospedar los servicios de Financial Business System en Windows Server AppFabric, debido a que es una tecnología que está más enfocada a la administración de servicios WCF, por lo tanto permite medir en forma más específica los detalles de cada servicio y también brinda la posibilidad de realizar una mayor cantidad de configuraciones sobre los mismos.
- Gestionar procesos de monitoreo constante sobre los sistemas de información con el fin de encontrar oportunidades de mejora en la arquitectura o en la configuración de las tecnologías utilizadas.

REFERENCIAS BIBLIOGRÁFICAS

Díaz, M., Garrido, D., & Troya, J. (2009). *Developing a communications architecture based on WCF for use in nuclear power plants simulators*. Universidad de Málaga, Málaga.

Lowy, J., & Michael, M. (2015). *Programming WCF Services* (4 ed.). O'Reilly Media.

Pathak, N. (2011). *Pro WCF 4 Practical Microsoft SOA Implementation*. Apress.

Sharp, J. (2010). *Windows Communication Foundation 4 Step by Step*. Microsoft.

Tharun Kola Siva, Prudhvi Muppalla, Reddy Satti Swamy. *Advantages of WCF Over Web Services*. En: International Journal of Computer Science and Mobile Computing. No 2. Abr., 2013); ISSN 2320-088X

Beroux Werner, *renameit* [en línea]. <https://github.com/wernight/renameit/> [citado en 8 de abril de 2016]

Beroux Werner, *renameit* [en línea]. <http://www.beroux.com/english/software/renameit/> [citado en 3 de mayo de 2016]

Microsoft Corporation, *Visual Studio Macros* [en línea]. [https://msdn.microsoft.com/en-us/library/b4c73967\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/b4c73967(v=vs.100).aspx) [citado en 3 de mayo de 2016]

Microsoft Corporation, *Automating Repetitive Actions by Using Macros* [en línea]. [https://msdn.microsoft.com/en-us/library/8h31zbch\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/8h31zbch(v=vs.100).aspx) [citado en 4 de marzo de 2016]

SifizSoft s.a., *Financial Business System* [en línea]. <http://www.sifizsoft.com/financial-business-system> [citado en 3 de mayo de 2016]

Microsoft Corporation, *What's New in Web Services Enhancements (WSE) 3.0* [en línea] <https://msdn.microsoft.com/en-us/library/ms977317.aspx> [citado en 3 de mayo de 2016]