# Memoria PFC .NET

Proyecto Fin de Carrera - Ingeniería Informática

Área: .NET

Autor: Oscar Prol Bravo Tutor: Xavier Navarro Esteve 12/06/2011



"A mí pareja, Cristina, por soportar de nuevo estar con un universitario."

# ÍNDICE DE CONTENIDOS

ÍNDICE	E DE CONTENIDOS	3
1 II	ntroducción	6
1.1.	Resumen Ejecutivo	6
2 [	Descripción del Proyecto	6
2.1.	Objetivos del Proyecto	6
2.2.	Resultados Esperados	7
2.3.	Ámbito del Proyecto	8
2.4.	Análisis de Riesgos	8
3 0	Organización del Proyecto	10
3.1.	Relación de Actividades	10
3.2.	Work BreakDown Structure	11
3.3.	Detalle de las Actividades	12
	Planificación del proyecto:	12
	Análisis de requisitos:	12
	Documentación (Estudio comparativa):	12
	Diseño:	13
	Implementación:	13
	Memoria y Presentación:	13
3.4.	Calendario de Trabajo	14
3.5.	Calendario de Trabajo Real	15
3.6.	Hitos Principales	16
3.7.	Equipo de Trabajo	17
3.8.	Definición de Roles	17
4 E	STUDIO	18
4.1.	Introducción	18
4.2.	SOA	19
4.3.	SERVICIOS WEB	21

	4.4.	WCF	23
	4.5.	COMPARATIVA: WS vs WCF	26
5	ANA	ÁLISIS	29
	5.1.	Análisis de Requisitos Funcionales	29
	5.2.	Modelo Conceptual	30
	5.3.	Actores	32
	5.4.	Diagrama de Casos de Uso	33
	5.4.1.	Casos de Uso del Usuario	34
	5.4.2.	Casos de Uso del Usuario Registrado	37
	5.4.3.	Casos de Uso del Comprador	38
	5.4.4.	Casos de Uso del Vendedor	39
	5.4.5.	Casos de Uso del Sistema	41
6	DIS	EÑO	42
	6.1.	Diseño Arquitectura Sistema	42
	6.1.1.	Arquitectura Actual del Sistema	42
	6.1.2.	Arquitectura del Nuevo Sistema	43
	6.2.	Diagrama de Clases	47
	6.2.1.	Entidades	47
	6.2.2.	Interfaces	53
	6.3.	Diagrama de Actividad	57
	6.4.	Diseño de la Base de Datos	59
	6.4.1.	Modelo Relacional	59
	6.5.	Diseño Interfaz Gráfica	60
	6.5.1.	Interfaz Autenticar Usuario	60
	6.5.2.	Interfaz Registrar Usuario	60
	6.5.3.	Interfaz Listado Artículos	61
	6.5.4.	Interfaz Detalle Artículo	61
	6.5.5.	Interfaz Confirmar Puja	62
	6.5.6.	Interfaz Crear Subasta Artículo	62
	6.5.1.	Interfaz Comentarios	62
7	' Mai	nual de Instalación	63

8	Manual de Usuario	. <b>72</b>
9	Objetivos Conseguidos en el Proyecto	. <b>72</b>
10	Conclusiones	. 88
11	BIBLIOGRAFÍA	. <b>90</b>
1:	1.1. Libros	90
1:	1.2. Enlaces Web	90

# **CAPÍTULO 1**

# 1 Introducción

# 1.1. Resumen Ejecutivo

El proyecto desarrolla un análisis de dos arquitecturas Orientadas a Servicios, como son los Servicios Web XML y WCF. La estructura del proyecto se divide en el estudio de las tecnologías y en la implementación de aplicaciones con cada una de ellas.

La planificación del proyecto se realiza a partir de los hitos externos que son las diferentes entregas de las pruebas de evaluación continua, así como la entrega final del proyecto.

Los resultados esperados en concreto son el estudio de las dos arquitecturas tecnológicas, así como la gestión completa de un proyecto.

# **CAPÍTULO 2**

# 2 Descripción del Proyecto

# 2.1. Objetivos del Proyecto

Este Proyecto de fin de carrera pretende desarrollar una comparativa entre Arquitecturas Orientadas a Servicios (SOA), en concreto se evaluarán la arquitectura WCF (Windows Comunitation Foundation) y los Servicios Web XML. Este proyecto se iniciará partiendo de un análisis técnico de las arquitecturas propuestas, para finalizar con la implementación de una misma aplicación con las dos tecnologías.

Los dos principales objetivos del proyecto son:

- Estudio comparativo entre las arquitecturas SOA siguientes :
  - o Arquitectura WCF
  - o Servicios Web XML de ASP.NET
- Implementación una aplicación con ASP.NET que utilice los Servicios Web y posteriormente se implementará una con WCF.

La aplicación que se desarrollará en este proyecto es un servicio de subastas. La implementación de la aplicación surge de la necesidad de una empresa de subastas en ofrecer sus servicios a otras compañías. De esta forma, la empresa abriría una nueva línea de mercado B2B. La aplicación tendrá que ofrecer todos los mecanismos necesarios para que otras empresas puedan crear y gestionar las subastas integrando este servicio en sus páginas web o en cualquier otro medio (dispositivos móviles, aplicaciones de escritorio, etc...).

La aplicación se implementará con las tecnologías que ofrece Microsoft en su Framework .NET, así pues el desarrollo se realizará con el entorno Microsoft Visual Studio, y el lenguaje de programación será C#.

# 2.2. Resultados Esperados

Los resultados que se pretenden conseguir una vez finalizado este proyecto de fin de carrera son los siguientes:

- Un correcto desarrollo en la gestión del proyecto aplicando los procedimientos y metodologías aprendidas en la carrera.
- La compresión y análisis en detalle de un tema tan actual como es la arquitectura orientada a servicio. Conseguir distinguir entre dos tecnologías como son los Web Services y WCF, pudiendo evaluar sus ventajas y desventajas de uno frente al otro.
- La implementación mediante una lenguaje de última generación como es C#, así como el uso de un Entorno de desarrollo integrado (IDE) tan

extendido como es Microsoft Visual Studio 2010. Como servidor de base de datos se utilizará Microsoft SQL Server, y para su acceso la tecnología ADO.NET. El tema de esta implementación será un servicio de subastas.

- La perfecta ejecución de todas las fases del proyecto.

# 2.3. Ámbito del Proyecto

El proyecto tiene como alcance la presentación de una documentación amplia sobre las tecnologías Web Services XML y WCF, donde se evaluarán y se definirán en detalle.

La implementación de dos prototipos será el acompañamiento del análisis realizado previamente, donde se realizara la muestra práctica de los conceptos explicados en la fase anterior.

Finalmente, se entregara un informe completo de todo el proyecto, memoria, donde se recogerá toda la documentación realizada. Junto con este informe se entregará un video presentación del proyecto, en el cual se defenderá el proyecto.

# 2.4. Análisis de Riesgos

Una vez finalizado el proyecto se realiza la valoración de los riesgos planificados, estos fueron definidos como posibles incidentes que pudieran afectar a la correcta ejecución del proyecto.

La valoración de los riesgos es muy positiva, ya que la ejecución del proyecto se ha realizado con la normalidad planificada. El riesgo 1, gestión y ejecución de otros proyectos en el mismo intervalo de tiempo ha sido mitigado correctamente, al igual que el riesgo 2.

Los riesgos 3 y 4 detallados en una primera planificación han sido mitigados según los planes de acción establecidos.

Riesgo 1	Nombre del Riesgo
Descripción	Gestión y ejecución de otros proyectos en el mismo intervalo de
	tiempo
Impacto	Medio
Probabilidad Muy Probable	
Acción de Mitigación	Correcta coordinación entre el proyecto de fin de carrera y el resto
	de asignaturas cursadas en el semestre. Otra medida será ampliar
	el calendario de trabajo semanal a sábados y Domingos.

Riesgo 2	Nombre del Riesgo
Descripción	Ejecución correcta del proyecto de fin de carrera.
Impacto	Alto
Probabilidad	Poco Probable
Acción de Mitigación	Mantener feedback continuo con el tutor del proyecto.

Riesgo 3	Nombre del Riesgo
Descripción	Desconocimiento de las arquitecturas SOA, en concreto Web Services y WCF.
Impacto	Medio
Probabilidad	Muy Probable
Acción de Mitigación	Realizar un estudio exhaustivo de las tecnologías en la fase de análisis.

Riesgo 4	Nombre del Riesgo	
Descripción	Calendario ajustado para la implementación de una aplicación con	
	tecnología Web Services y otra con tecnología WCF.	
Impacto	Alto	
Probabilidad	Muy Probable	
Acción de Mitigación	Extensa documentación en la fase análisis de la aplicación para que el tiempo de implementación sea el mínimo posible y se reduzca considerablemente el riesgo. Así como la correcta definición de los requisitos para que el prototipo contenga únicamente lo necesario para el propósito del proyecto.	

# **CAPÍTULO 3**

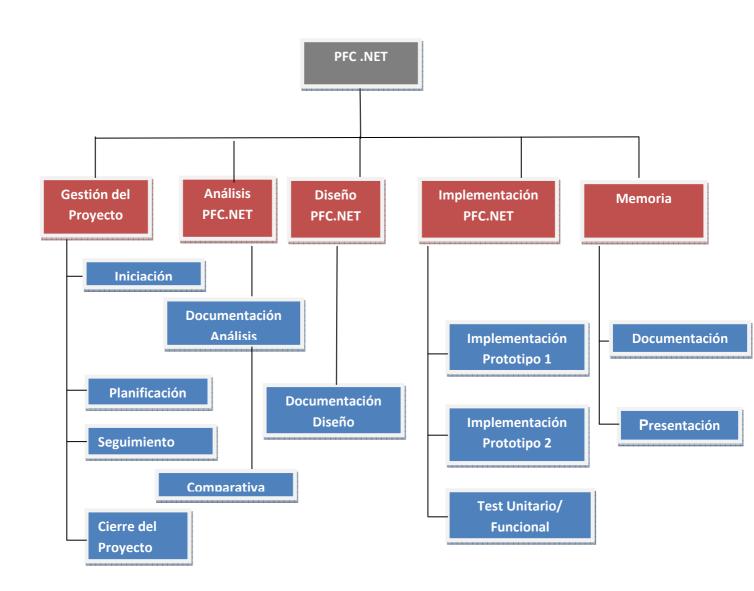
# 3 Organización del Proyecto

# 3.1. Relación de Actividades

Las actividades de este proyecto son las siguientes:

- Planificación del Proyecto
- Análisis de Requisitos
- Documentación
- Diseño
- Implementación
- Memoria y Presentación

# 3.2. Work BreakDown Structure



WBS Nivel 1	WBS Nivel 2	WBS Nivel 3	%
PFC .NET			100%
	Gestión del Proyecto		5%
		Iniciación	1%
		Planificación	1%
		Seguimiento	1.5%
		Cierre del Proyecto	1.5%
	Análisis PFC.NET		30%
		Documentación Análisis	15%
		Estudio Comparativa	15%
	Diseño PFC.NET		15%
		Documentación Diseño	15%
	Implementación PFC.NET		40%
		Implementación Prototipo	25%
		1	
		Implementación Prototipo	10%
		2	
		Test Unitario y Funcional	5%
	Memoria		10%
		Documentación	5%
		Presentación	5%

# 3.3. Detalle de las Actividades

A continuación se detalla cada una de las actividades:

Planificación del proyecto: Esta actividad identificará todo el ámbito del proyecto así como el tiempo y los recursos que se definen en este. Como artefacto de esta fase surge el presente documento de planificación del proyecto de fin de carrera.

Análisis de requisitos: Esta actividad identificará las metas globales con la definición de todas las necesidades y requerimientos del proyecto en cuestión. De esta fase surgirá como artefacto el diagrama de casos de uso.

Documentación (Estudio comparativa): Esta actividad tendrá como resultado un análisis detallado de las dos arquitecturas evaluadas en este proyecto, la arquitectura WCF y los Servicios Web. Como artefacto de esta fase surgirá el análisis y las conclusiones extraídas del estudio de estas dos arquitecturas.

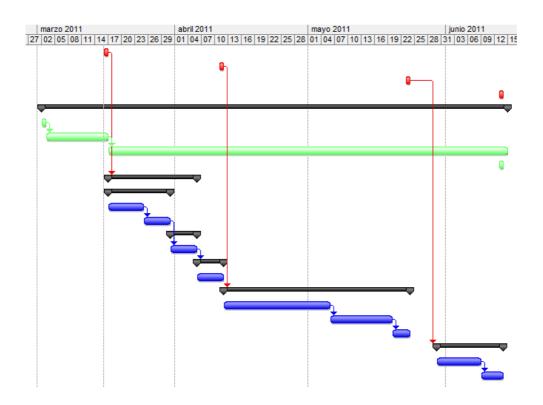
Diseño: Esta actividad definirá los requerimientos de la fase anterior de análisis y comprenderá todos los diseños de arquitectura, de datos, de interfaces y de los procedimientos. En el diseño de los datos se mostraran las estructuras de datos necesarias para la implementación del prototipo, en el diseño de la arquitectura se mostrarán los elementos estructurales del programa, en el diseño de interfaces se describirá la comunicación con los usuarios y finalmente en el diseño de procedimientos la transformación de los elementos estructurales. De esta fase surgirán varios artefactos entre los que se destacan el diagrama de clases, el diagrama de persistencia y el diagrama de colaboración.

Implementación: Esta actividad corresponde a la construcción de los dos prototipos en base a la fase previa de análisis. Un prototipo se implementará por medio de Web Services mientras que el otro utilizará la tecnología WCF. De esta fase surgirá como artefactos los dos prototipos así como el documento de manual de instalación y manual de usuario.

Memoria y Presentación: Esta actividad trata de realizar el documento que resumirá todo el proceso del proyecto, memoria, desde su planificación, análisis y diseño hasta la implementación y la documentación de uso. A su vez, esta fase estará acompañada de un video donde se defenderá el presente proyecto, por lo tanto deberá tener una presentación del trabajo realizado, así como muestras de ejecución del resultado del proyecto.

# 3.4. Calendario de Trabajo

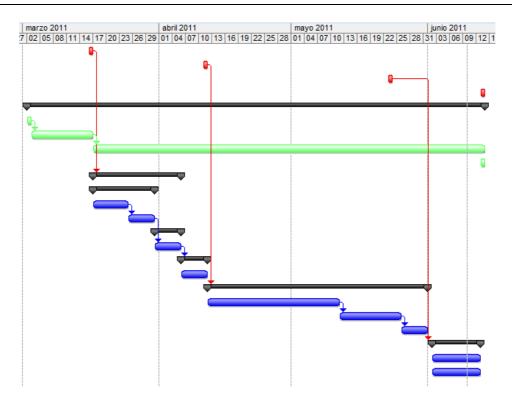
	0	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1	<b>III</b>	Hito Externo : PAC 1	1 día?	mié 16/03/11	mié 16/03/11	
2	1	Hito Externo : PAC 2	1 día?	lun 11/04/11	lun 11/04/11	
3	1	Hito Externo : PAC 3	1 día?	lun 23/05/11	lun 23/05/11	
4	1	Hito Externo : Entrega Final	1 día?	lun 13/06/11	lun 13/06/11	
5		□ Planificación del Proyecto	75 días?	mié 02/03/11	mar 14/06/11	
6		Iniciación	1 día?	mié 02/03/11	mié 02/03/11	
7	1	Planificación	10 días?	jue 03/03/11	mié 16/03/11	6
8	<b>III</b>	Seguimiento	64 días?	jue 17/03/11	mar 14/06/11	7
9	===	Cierre	1 día?	lun 13/06/11	lun 13/06/11	
10		─ Análisis de Requisitos	14 días	jue 17/03/11	mar 05/04/11	1
11		─ Estudio Comparativa	10 días	jue 17/03/11	mié 30/03/11	
12		Realizar Estudio Arquitecturas	6 días	jue 17/03/11	jue 24/03/11	
13		Creación del Documento	4 días	vie 25/03/11	mié 30/03/11	12
14		Análisis	4 días	jue 31/03/11	mar 05/04/11	
15	===	Creación del Documento	4 días	jue 31/03/11	mar 05/04/11	13
16		□ Diseño	4 días	mié 06/04/11	lun 11/04/11	15
17		Creación del Documento	4 días	mié 06/04/11	lun 11/04/11	
18		☐ Implementación	30 días	mar 12/04/11	lun 23/05/11	2
19		Prototipo 1	18 días	mar 12/04/11	jue 05/05/11	
20		Prototipo 2	10 días	vie 06/05/11	jue 19/05/11	19
21		Test Funcionales	2 días	vie 20/05/11	lun 23/05/11	20
22		□ Memoria	11 días	lun 30/05/11	lun 13/06/11	3
23	<b>III</b>	Creación del Documento	8 días	lun 30/05/11	mié 08/06/11	
24	===	Creación de la Presentación	3 días	jue 09/06/11	lun 13/06/11	23



# 3.5. Calendario de Trabajo Real

El calendario de trabajo real se ha mantenido perfectamente durante todo el proyecto hasta la fase final, ya que para cumplir con el hito externo PAC3, se han tenido que añadir más días de trabajo de los estimados. La ampliación del tiempo se ha realizado incorporando los cuatro últimos fines de semana al calendario de trabajo, ya que no estaban considerados los fines de semana como días de trabajo en la planificación inicial.

	Nombre de tarea	Duración	Comienzo	Fin
1	Hito Externo : PAC 1	1 día?	mié 16/03/11	mié 16/03/11
2	Hito Externo : PAC 2	1 día?	lun 11/04/11	lun 11/04/11
3	Hito Externo : PAC 3	1 día?	lun 23/05/11	lun 23/05/11
4	Hito Externo : Entrega Final	1 día?	lun 13/06/11	lun 13/06/11
5	□ Planificación del Proyecto	75 días?	mié 02/03/11	lun 13/06/11
6	Iniciación	1 día?	mié 02/03/11	mié 02/03/11
7	Planificación	10 días?	jue 03/03/11	mié 16/03/11
8	Seguimiento	64 días?	jue 17/03/11	lun 13/06/11
9	Cierre	1 día?	lun 13/06/11	lun 13/06/11
10	─ Análisis de Requisitos	14 días	jue 17/03/11	mar 05/04/11
11	─ Estudio Comparativa	10 días	jue 17/03/11	mié 30/03/11
12	Realizar Estudio Arquitecturas	6 días	jue 17/03/11	jue 24/03/11
13	Creación del Documento	4 días	vie 25/03/11	mié 30/03/11
14	- Análisis	4 días	jue 31/03/11	mar 05/04/11
15	Creación del Documento	4 días	jue 31/03/11	mar 05/04/11
16	⊡ Diseño	4 días	mié 06/04/11	lun 11/04/11
17	Creación del Documento	4 días	mié 06/04/11	lun 11/04/11
18	☐ Implementación	36 días	mar 12/04/11	mar 31/05/11
19	Prototipo 1	22 días	mar 12/04/11	mié 11/05/11
20	Prototipo 2	10 días	jue 12/05/11	mié 25/05/11
21	Test Funcionales	4 días	jue 26/05/11	mar 31/05/11
22	■ Memoria	8 días	jue 02/06/11	dom 12/06/11
23	Creación del Documento	8 días	jue 02/06/11	dom 12/06/11
24	Creación de la Presentación	8 días	jue 02/06/11	dom 12/06/11



# 3.6. Hitos Principales

Los hitos externos del proyecto son los siguientes:

Fecha	Descripción del Hito
16/03/2011	Entrega PAC 1
11/04/2011	Entrega PAC 2
23/05/2011	Entrega PAC 3
13/06/2011	Entrega Final

Los hitos internos del proyecto son los siguientes:

Fecha	Descripción del Hito
30/03/2011	Creación Documento Estudio Tecnologías
05/04/2011	Creación Documento Análisis
11/04/2011 Creación Documento Diseño	
05/05/2011 Creación Prototipo 1	
19/05/2011	Creación Prototipo 2
08/06/2011	Creación Documento Memoria
13/06/2011	Creación Presentación Memoria

# 3.7. **Equipo de Trabajo**

Componente	Rol	
Oscar Prol Bravo	Gestor	
	Analista	
	Responsable Desarrollo	

# 3.8. **Definición de Roles**

Rol	Descripción
Gestor	Responsable de supervisar globalmente el proyecto y controlar la ejecución de este, asegurando el cumplimiento de los objetivos y del tiempo.
Analista	Responsable de la recogida de requisitos, así como el diseño de la solución.
Responsable Desarrollo	Responsable de la implementación de los prototipos.

# **CAPÍTULO 4**

# 4 ESTUDIO

## 4.1. Introducción

Los paradigmas de la programación han evolucionado durante las tres últimas décadas, implementando diferentes tendencias con el mismo fin, la ayuda a la programación en la construcción del software con un foco principal, reutilización del código. En las tres últimas décadas se puede resumir los movimientos en la programación orientada a objetos (década de los 80), la programación orientada al componente (década de los 90) y la programación orientada a servicio (década del 2000). Cada uno de los movimientos mencionados aporta a la programación de software diferentes características. A continuación se detallan las principales de cada tendencia y como cada una de ellas complementa las carencias o defectos del movimiento anterior.

Programación Orientada a Objeto: Ofrece la posibilidad de desarrollar construyendo por medio de clases reusables, que permiten la herencia de todas las funcionalidades de sus clases base. Fue un avance sustancial a la programación por procedimientos que existía hasta el momento, ya que aportaba el polimorfismo, la abstracción y la encapsulación. Por otro lado, no facilitan la evolución del software en ejecución, ya que la una vez creada la aplicación esta es estática. Sin facilitar la incorporación o ampliación del código en la aplicación.

Programación Orientada a Componentes: Ofrece la posibilidad de sobreponer el problema de la aplicación estática que tenía la programación Orientada a Objetos. Este paradigma obliga al programador en diseñar siempre interfaces externos en la aplicación. Como beneficios de este movimiento son, el poder permitir construir sistemas dinámicamente y unirlos en tiempo de ejecución. La carga de los componentes es dinámica, en tiempo de ejecución ampliando la capacidad de la aplicación. Pero el problema de esta tendencia es cuando se quiere aplicar el concepto a diferentes maquinas o sistemas, porque aunque funcione perfectamente en un único sistema, se tienen problemas al intentar integrar con sistemas distribuidos.

Programación Orientada a Servicios: Ofrece las facilidades de encapsulación y la funcionalidad de carga dinámica. En esta tendencia cambia los problemas al integrar con sistemas distribuidos. Este problema lo soluciona por medio de comunicarse mediante mensajes en lugar de realizar invocaciones a referencias de objetos. Por lo tanto el principal beneficio que ofrece es la integración del software en un sistema totalmente escalable y distribuido. A demás para suportar la carga de funcionalidades dinámicamente, utiliza un esquema que describe la estructura de los mensajes, el comportamiento del contrato para definir patrones de intercambio de mensajes y la política de definir la semántica de los servicios.

En la siguiente figura se pueden ver la transición de los movimientos mencionados en la línea del tiempo.

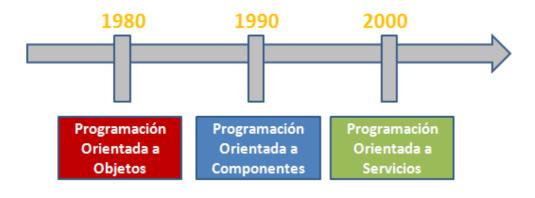


Ilustración 1

El foco de este estudio se centra en la Arquitectura Orientada a Servicio, principalmente se detallará el principal estándar de esta arquitectura como son los Servicios Web y por otra parte la plataforma Windows Communication Foundation (WCF), colección de interfaces ofrecida por Microsoft en su Framework dot Net.

# 4.2.**SOA**

Arquitectura Orientada al Servicio (SOA) es una colección de Servicios bien definidos, donde cada Servicio individual puede ser modificado independientemente de los otros servicios para ayudar a responder a una evolución del sistema. La diferencia más importante que provee SOA respecto a las predecesoras tecnologías (programación orientada a componentes), es el uso de mensajes para la comunicación, y que estos están definidos mediante estándares abiertos.

El uso de los estándares abiertos en la definición de toda la tecnología hace que sea genérica, y que no forme parte de ninguna plataforma en específico ni de ningún lenguaje de

programación. Cada uno de estos servicios es autónomo y provee al sistema de varios métodos o funcionalidades, por detrás queda oculta la implementación, lo que hace que cualquier cambio sea transparente para el consumidor del servicio. Siempre y cuando no cambie el contrato. Esta característica proviene del paradigma de la programación orientada a objetos, la encapsulación.

La Orientación a Servicio es una estrategia que define las funcionalidades de un negocio en términos de sistemas autónomos débilmente acoplados que intercambian información entre ellos por medio de mensajes. Los cuatro principios del Servicio son:

- Los límites están explícitos: Especifica la publicación y consumo de las funcionalidades de varios servicios sin conocer la implementación de estos, encapsulación.
- Los Servicios son autónomos: Las evoluciones que sufra la implementación del servicio a lo largo del tiempo tiene que ser soportada por el diseño del sistema.
- Los Servicios comparten Esquemas y Contratos, no clases: El Servicio publica el contrato que los clientes usaran para generar las invocaciones al Servicio y poder consumirlo. No se compartirán tipos entre el Servicio y los clientes por defecto. El consumidor únicamente con el esquema y el contrato podrá hacer uso del Servicio.
- La compatibilidad del Servicio se determina en base a la política: El Servicio puede negociar los canales de comunicación dinámicamente, siempre y cuando soporten las semánticas necesarias de seguridad y fiabilidad del sistema. Las políticas del Servicio son creadas automáticamente en la configuración, atributos de clases y métodos de firma. El canal del cliente o consumidor son configurados automáticamente al recibir las políticas del Servicio consumido.

Los beneficios que pueden obtenerse de la implementación de la Arquitectura Orientada a Servicio son:

- La implementación de cambios en los Servicios de un sistema es sustancialmente inferior.
  - Mejora los cambios de evolución en las comunicaciones a terceros.
  - La colaboración con otros sistemas es simple y manejable.
  - El Sistema no se ve interrumpido por cambios en capas inferiores.
  - La integración con tecnologías distintas es más sencillo.

Como principal alternativa para la implementación de una Arquitectura Orientada a Servicio se encuentran los Servicios Web, implementando los protocolos SOAP y WSDL, aunque se puede implementar la arquitectura con otras tecnologías.

#### 4.3. **SERVICIOS WEB**

Los servicios Web son una colección de protocolos y estándares, que se usan para el intercambio de información entre sistemas. Como principal característica, estos servicios están publicados, definidos y localizados en Internet. Las características que tienen los Servicios Web son las siguientes:

- Los Servicios Web son accesibles por la Web de Internet.
- La comunicación con los Servicios Web es independiente de la plataforma, ya que se usa protocolos con un lenguaje neutral.
- El Servicio Web esta registrado y puede ser localizado por medio del registro de Servicios Web.
- El Servicio Web permite la conexión entre sistemas bajamente acoplados.

Los aspectos importantes de los Servicios Web son los siguientes:

- Protocolo Estándar: El Servicio es expuesto funcionalmente vía interfaces usando los principales protocolos de Internet como son HTTP, SMTP, FTP, etc.... Aunque en la mayoría de casos el protocolo es el HTTP.
- Descripción del Servicio: Los Servicios Web necesitan describir todas sus interfaces en detalle para que el cliente que quiera consumirlo disponga de toda la funcionalidad dispuesta por el Servicio. Esta descripción se realiza en XML, mediante un documento llamado WSDL (Web Services Description Language).
- Búsqueda de Servicios: Los clientes necesitan saber que Servicios existen y donde esta ubicados para poder consumirlos. Para que los clientes conozcan y encuentren los Servicios disponen de un repositorio UDDI (Universal Discovery, Description and Integration), donde se muestran el listado de Servicios disponibles.

Por lo tanto los protocolos que utiliza el Servicio Web son las siguientes:

- XML (eXtensible Markup Language) :
- SOAP (Simple Object Access Protocol) :
- WSDL (Web Services Description Language):
- UDDI (Universal Description, Discovery, and Integration):

XML es un metalenguaje extensible de etiquetas desarrollado por W3C (World Wide Web Consortium). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos. Los Servicios Web utilizan este protocolo para representar la información, esta información se estructurará mediante los esquemas de SOAP.

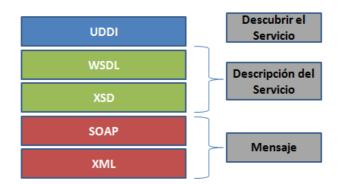
SOAP es un protocolo de comunicaciones para los Servicios Web que se basa en el estándar XML. Este protocolo se usa para el intercambio de información de forma estructurada y tipificada entre los sistemas. SOAP permite la invocación de métodos en diferentes sistemas sin tener que conocer los detalles de su plataforma o aplicaciones que se están ejecutando. La comunicación SOAP se realiza mediante los Mensajes SOAP que contienen la siguiente información:

- Primera Parte: SOAP envuelve el documento XML que encapsula el mensaje que se está comunicando.
- Segunda Parte: Se usa para definir tipos de datos personalizados que existen en el Servicio.
- Tercera Parte: Define el patrón RPC (Remote Procedure Call) que va a ser usado.
- Cuarta Parte: Define como SOAP se une a HTTP.

WSDL es un lenguaje basado en XML para describir el comportamiento de un Servicio Web, conteniendo los métodos y los tipos de datos necesarios para definir dichas operaciones. Es necesario para que el cliente o consumidor pueda invocar el Servicio Web.

UDDI es un registro independiente de la plataforma basado en XML para registrar los Servicios Web y permitir posteriormente descubrirlos.

En la siguiente figura se puede ver la pila de protocolos para los Servicios Web.



#### Ilustración 2

Para consumir un Servicio Web primero se debe descubrir el Servicio y ver que ofrece, esta operación se realiza mediante el protocolo UDDI. Una vez, se haya descubierto el Servicio se debe entender la interface, es decir, los métodos y parámetros que acepta el Servicio para ser consumido. Esta operación se realiza mediante los protocolos WSDL y XSD. Finalmente el consumo del Servicio Web se realiza mediante mensajes los cuales están en formato XML y encapsulados mediante el protocolo SOAP. De esta forma y en este orden, se utilizaría la pila de protocolos de un Servicio Web. En la siguiente figura se muestra el ciclo de vida del Servicio Web:

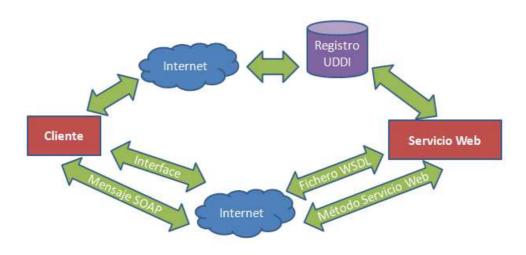


Ilustración 3

# 4.4. WCF

Windows Communication Foundation (WCF) es una interface para desarrollar aplicaciones distribuidas por medio del modelo de programación Orientado a Servicio. Este interface de herramientas está incluido dentro del Framework de trabajo ofrecido por Microsoft, Framework dot NET. Este FrameWork dot NET es un gran conjunto de librerías y soporte para variados lenguajes que permite la interoperabilidad entre estos.

Este Framework de librerías de Microsoft en su versión 3, incluye el denominado Windows Communication Framework.

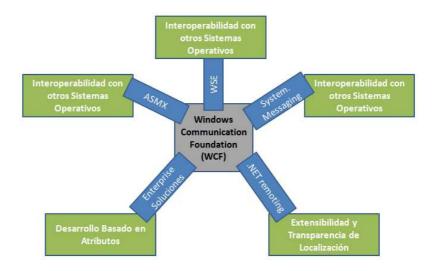
WCF está diseñado de acuerdo a los principios de la Arquitectura Orientada a Servicios, para soportar programación distribuida donde los Servicios son consumidos por los clientes y estarán débilmente acoplados.

WCF soporta interoperabilidad entre las aplicaciones desarrolladas con esta API (Application Programming Interface) en la mismas maquinas o ejecutadas en diferentes máquinas Windows, o Servicios Web estándar construidos en otras plataformas como puede ser Java, ejecutándose en diferentes sistemas operativos. WCF no solo soporta los mensajes SOAP, si no que puede ser configurado para trabajar con el estándar XML sin tener que ser envuelto por SOAP. También soporta una gran variedad de formatos como pueden ser RSS o JSON que hacen más flexible a WCF para los requerimientos actuales, así como futuros.

WCF ha sido desarrollado con tres claros objetivos: La productividad, la Interoperabilidad y Orientación a Servicio.

La productividad la consigue disponiendo de varias tecnologías para el desarrollo de aplicaciones distribuidas (COM+ and .NET Entreprise Services, MSMQ, .NET Remoting, ASP.NET Web Services y Web Services Enhacements(WSE)), en un solo y unificado modelo de

programación, unificación de las tecnologías existentes. Este modelo reduce la complejidad de aprender un único modelo en lugar de todos los mencionados. Otro beneficio que aporta es la combinación de funcionalidades de las diferentes tecnologías. Y por último, permite usar un modelo de programación para desarrollar aplicaciones distribuidas para comunicarse en un maquina, varias maquinas y en Internet. En la siguiente figura se muestran la unificación de las tecnologías distribuidas.



#### Ilustración 4

- ASMX es un Servicio Web que está disponible mediante estándares abiertos como son SOAP y HTTP.
- MSMQ (System.Messaging) es un protocolo de mensajes que permite a las aplicaciones distribuidas comunicarse por colas de mensajes.
- WSE es un Servicio Web que ofrece características extendidas sobre un Servicio Web como son el enrutamiento, la seguridad, mensajes confiables y transacciones.
- Enterprise Servicios es una infraestructura que ofrece accesos a servicios y ciertas características dentro del ámbito, para los desarrolladores que están construyendo la aplicación.

La Interoperabilidad la consigue construyendo servicios que usan protocolos Servicios Web avanzados (Ws-\*). Ofrecer disponible las comunicaciones con aplicaciones de diferentes plataformas con la suficiente flexibilidad que se precisa cuando se trata de un entorno heterogéneo. La compatibilidad con las aplicaciones existentes que ofrece una transición gradual a WCF.

Orientado a Servicio, ofrece un alto modelo de desarrollo productivo para la construcción de estas aplicaciones.



Ilustración 5

En la figura anterior se muestran las principales aportaciones de la WCF.

La arquitectura de la WCF se resume en tres modelos: el modelo de desarrollo, el modelo de servicio y el modelo de canal. En la siguiente figura se muestran los diferentes modelos que definen la arquitectura de las WCF.

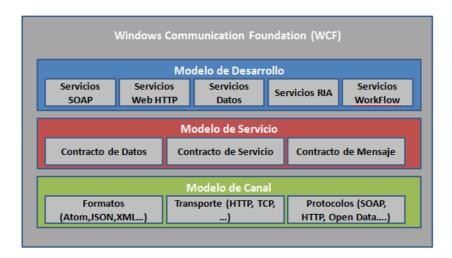


Ilustración 6

En el modelo de servicio están los contratos de datos, servicios y comportamiento que definirán varios aspectos del sistema de mensajes. El contrato de datos describe cada parámetro que constituye cada mensaje que un servicio puede crear o utilizar. El contrato de servicios especifica las firmas del método actuales del servicio y se distribuye como una interfaz en uno de los lenguajes de programación compatibles, como VB o VC#. El contrato del mensaje define partes específicas del mensaje utilizando los protocolos SOAP y permite el control más fino sobre las partes del mensaje, cuando la interoperabilidad exige tal precisión.

En el modelo de canal es donde se establecen los componentes que procesan un mensaje. Existen dos tipos de canales, los de transporte y los de protocolo. Los canales de transporte leen y escriben mensajes de la red, por ejemplo HTTP, TCP o MSMQ. Los canales de protocolo implementan protocoles de procesamiento de mensajes, realizando lecturas o escrituras de los encabezados adicionales en el mensaje, por ejemplo WS-Security y WS-Reliabitility.

En el modelo de desarrollo son los diferentes tipos de servicios que se pueden generar, ya como un programa o como servicios hospedado, como la aplicación IIS (Internet Information Server) o WAS (Servicio de activación de Windows).

# 4.5. COMPARATIVA: WS vs WCF

WCF es parte del FrameWork dot NET en la versión 3 que provee de un modelo de programación unificada para implementar aplicaciones Orientadas a Servicio, mientras que los Servicios Web están desarrollados para construir aplicaciones para enviar y recibir mensajes por medio de SOAP sobre HTTP. Por lo tanto, no se pueden comparar directamente las dos tecnologías ya que ofrecen la misma solución para el desarrollo de entornos distribuidos, aunque WCF extiende su ámbito más que a un simple Servicio Web.

Los Servicios Web envían y recibe mensajes SOAP sobre HTTP, la estructura de los mensajes está definida en un esquema XML. La tecnología puede generar metadatos automáticamente para describir los Servicios en la Web mediante WSDL.

WCF es una plataforma que permite enviar mensajes a las aplicaciones implementadas con el Framework dot NET con otros sistemas. SOAP es por defecto el protocolo utilizado, aunque se pueden definir los mensajes en cualquier formato y usar diferentes protocolos de transporte. La estructura de los mensajes está definida también en un esquema XML. Al igual que los Servicios Web también puede generar metadatos automáticamente para describir los Servicios Web mediante WSDL.

WCF ofrece la posibilidad de integrar diferentes tecnologías como los Servicios Web, COM+, MSMQ y .NET Remoting en un mismo entorno de desarrollo.

Mientras que los Servicios Web se deben hospedar dentro de un Servidor Web como puede ser el IIS, WCF puede crear aplicaciones stand-alone o crear servicios que realicen el mismo propósito.

WCF permite datos más complejos, ya que serializa los datos por medio de "DatacontractSerializer" para sobrepasar los límites establecidos.

Las aplicaciones WCF pueden ser configuradas fácilmente por medio de ficheros de configuración y diferentes utilidades.

La principal característica de WCF es la integración con otras tecnologías y su gran interoperabilidad.

A continuación se detallan las diferentes tecnologías que soporta WCF, los Servicios Web son únicamente una de las tecnologías mostradas:

- Servicios Web

- Comunicación Binaria (.NET Remoting)
- Transacciones Distribuidas
- Soporte a Especificación WS-\*
- MSMQ (Colas de Mensajes)

Por otra parte las tres principales limitaciones que tienes los Servicios Web son que únicamente puede ser hospedadas en un Servidor Web, únicamente se puede consumir mediante HTTP y que la seguridad es muy limitada.

A continuación se muestra una tabla con los conceptos diferentes entre un Servicio Web y WCF.

Característica	ASP.NET Servicio Web	WCF
Transformación de los Datos	XMLSerializer realiza la transición de la información de entrada y salida.	DataContractSerializer se usa para la transición.
Extensión Archivo	La extensión es ASMX.	La extensión es SVC.
Métodos Web vs Atributos DataContract	Usa los Web métodos para traducir los tipos propios del .NET FW a XML.	Usa DataContractAttribute y DataMemberAttribute para traducir los tipos .NET FW a XML.
Limitaciones	Únicamente las clases que implementan el interface IEnumerable o ICollection pueden ser serializadas.	Los campos o propiedades de .NET pueden ser traducidos.
Clase IDictionary Interface	No se pueden serializar las clases que implementen IDictionary.	DataContractSerializer puede traducir una tabla hash en XML.
Rendimiento	Más Lento comparado con WCF.	El mayor beneficio del diseño de DataContractSerializer mejora el rendimiento.
Conocer las propiedades y tipos de campos	XMLSerialization no indica los tipos de campo que va a ser serializados a XML.	DataContractSerializer explícitamente muestra los campos que son serializados a XML.
Manejo de Excepciones	Las excepciones no manejadas son enviadas como Fallo SOAP.	No son enviadas como fallo SOAP, la configuración permite devolver las excepciones no manejadas para depurar.

Como característica a mencionar es el rendimiento de las WCF sobre los Servicios Web, según aporta Microsoft en su propia página web (véase: <a href="http://msdn.microsoft.com/en-us/library/bb310550.aspx">http://msdn.microsoft.com/en-us/library/bb310550.aspx</a> ), WCF mejora a ASMX en un 27%, 31% y 48% para 1, 10 y 100 objetos en el mensaje respectivamente. Es decir, sobre un test sobre la comunicación entre un

cliente y un servidor, donde se pasar un entero y un array de objetos (1,10 y 100). Los resultados se muestran en un procesador y en un quadprocessor. A parte de esta comparativa de rendimiento se realiza la misma comparación para el resto de tecnologías como .NET Remoting, MSMQ, etc.... siendo WCF quien ofrece mejor rendimiento.

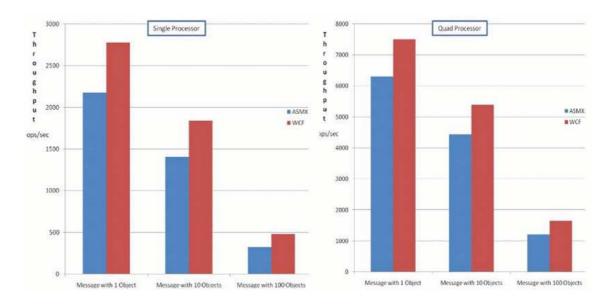


Ilustración 7

Con esta última característica de rendimiento se concluye aportando que WCF además de integrar varias tecnologías distribuidas, mejorar la productividad e aportar interoperabilidad entre plataformas, mejora el rendimiento de estas tecnologías.

# **CAPÍTULO** 5

# 5 ANÁLISIS

# 5.1. Análisis de Requisitos Funcionales

La necesidad de la compañía Subasta S.A., es abrir su mercado ofreciendo sus servicios en un entorno B2B. Para ello tiene la necesidad de ofrecer los mismos servicios que ofrece en su entorno Web, a terceros.

El modelo de negocio de la empresa de subastas actual es el siguiente:

Los usuarios pueden visualizar todas las subastas de artículos que hay en el sistema. La información que se muestra en el listado son los nombres de los artículos con una pequeña descripción que ofrece al usuario un poco más de información.

Los usuarios pueden acceder a cada artículo para visualizar los detalles de estos. En el detalle se muestra una fotografía, una descripción más extensa que detalla a la perfección el artículo subastado. En este apartado se debe ofrecer la posibilidad al usuario de ver los datos de la subasta, como son la fecha de inicio, la fecha de finalización, el precio actual del artículo, así como el detalle de las pujas que existen.

Las subastas se pueden ganar de dos maneras distintas, la primera por la finalización del tiempo teniendo en ese preciso momento la puja más alta en el artículo en cuestión. La segunda es realizar la compra del artículo mediante el precio que ha establecido el vendedor en el momento de la puesta en marcha de la subasta.

Los usuarios debidamente registrados podrán dar de alta artículos en el sistema de subastas, previo el pago del servicio que se está solicitando.

Cada artículo que esta dado de alta en el sistema para una subasta se encontrara debidamente categorizado.

Las subastas pueden tener tres estados, son los siguientes:

- Pendiente: La subasta ha sido registrada en el sistema, pero el pago del coste de esta no está pagado por el vendedor.
- Activa: La subasta está registrada, y dependiendo de la fechas de inicio aceptará las pujas por los compradores potenciales.
- Cerrada: Las subasta ha finalizado, ya sea porque la fecha de fin de la subasta ha sido superada o porque se ha producido la opción de comprar ahora.

Una vez finalizada la subasta tanto el vendedor como el comprador podrán ofrecer un comentario o feedback del otro, es decir, el vendedor podrá ofrecer su opinión de la visión que tiene del comprador según el tipo de respuesta del pago. Por otro lado, el comprador podrá ofrecer su opinión de la visión que tiene del vendedor referente al tiempo que ha tardado en recibir el artículo, así como el estado o la similitud de este a la descripción que tenía en la subasta.

El comentario puede ser únicamente: positivo, neutro o negativo.

La empresa Subastas S.A., realiza un cobro de un importe determinado por los usuarios que utilizan el sistema de subastas, es decir, cada usuario paga una cantidad determinada por cada subasta que realiza. Por otro lado, gestiona el cobro de la subasta pero únicamente como intermediario, ya que el precio total que paga el comprador por la subasta está destinado íntegramente al vendedor.

El principal requisito del cliente es ofrecer su sistema de subastas como servicio, la lógica de su negocio debe ser respetada. Se debe facilitar la interfaz necesaria para que las empresas que quieran trabajar con su sistema de subastas puedan hacerlo con total disponibilidad y transparencia para su sistema actual.

# 5.2. Modelo Conceptual

A partir del análisis de requisitos funcionales se extraen las siguientes entidades:

- Artículo
- Subasta
- Estado
- Puja
- Categoría
- Tipo de Comentario
- Comentario
- Usuario

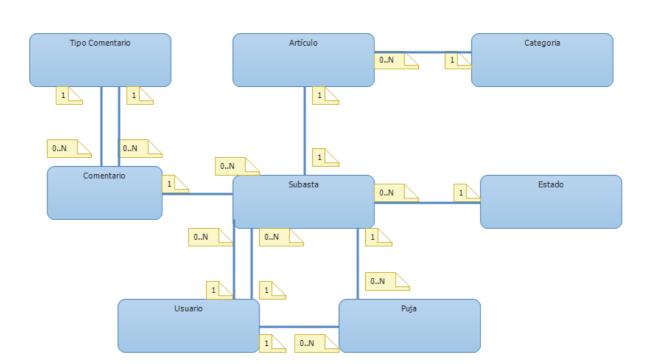


Ilustración 8

Un artículo pertenece a una categoría.

Un artículo se encuentra en una subasta y una subasta únicamente tiene un artículo.

La subasta posee un estado que la define.

Las subastas pueden tener varias pujas realizadas por los usuarios por un artículo en cuestión.

Los usuarios pueden vender, pujar o comprar una subasta.

Las subastas tienen comentarios de los compradores y vendedores.

Los comentarios tienen un tipo de comentario.

# 5.3. Actores

En el sistema se diferencian cuatro actores:

- Usuario: son las personas esporádicas o no registradas en el sistema que pueden visualizar todos los artículos subastados.
- Comprador: son las personas que realizan pujas sobre artículos. Están debidamente registrados en el Sistema. Son compradores potenciales de los artículos.
- Vendedor: son las personas que crean subastas con sus artículos.
   Están debidamente registrados en el Sistema.
- Sistema: se encarga del cierre de las subastas.

Como existen algunas funcionalidades que son comunes entre los compradores y los vendedores, ya que son usuarios registrados que pueden realizar o actuar de las dos formas. Por lo tanto se extrae la creación de un actor Usuario Registrado del cual heredaran el Comprador y el Vendedor las funcionalidades comunes.

A su vez el actor Usuario Registrado hereda los casos de uso del Usuario.

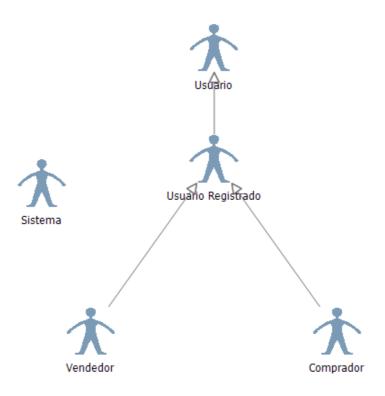


Ilustración 9

# 5.4. Diagrama de Casos de Uso

El Diagrama general de Casos de uso es el siguiente:

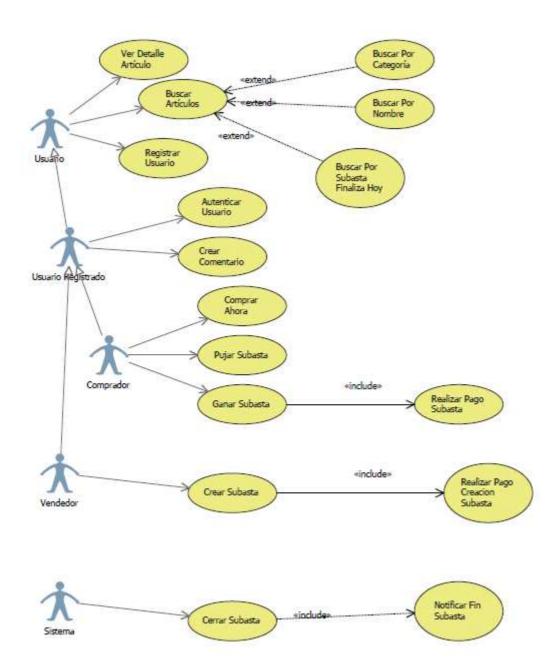


Ilustración 10

#### 5.4.1.Casos de Uso del Usuario

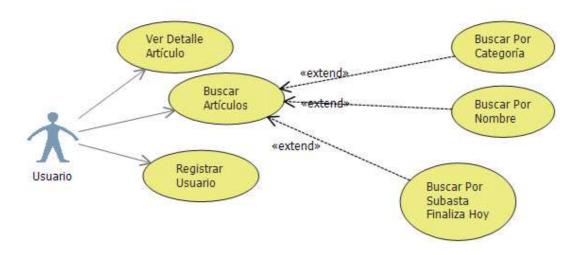


Ilustración 11

# Código: CU1

**Nombre: Registrar Usuario** 

Descripción: Permite registrar al usuario en el sistema.

Actor: Usuario

Precondiciones: No aplicable.

Postcondiciones: El usuario ha sido registrado en el sistema.

# Flujo:

- 1.- El Sistema solicita los datos de registro del Usuario: (NickName,Password,Nombre, Dirección,etc...)
- 2.- El Usuario introduce los datos requeridos.
- 3.- El Sistema valida los datos introducidos por el Usuario.
- 4.- El Sistema contesta con la validación de los datos introducidos por el Usuario.
- 5.- El Sistema contesta con la validación correcta de los datos introducidos por el Usuario.

# Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU2

## Nombre: Buscar Artículos

**Descripción:** Permite listar un conjunto de artículos en subasta.

Actor: Usuario y Usuario Registrado

**Precondiciones:** El Usuario Registrado ha sido autenticado en el sistema. **Postcondiciones:** El usuario ha consultado la lista de Artículos solicitada.

#### Flujo:

- 1.- El Usuario solicita la lista de artículos subastados.
- 2.- El Sistema devuelve la información de los artículos subastados.
- 3.- El Usuario consulta la lista de Artículos en subasta.

## Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU2.1

#### **Nombre: Buscar Artículos Por Nombre**

**Descripción:** Permite listar un conjunto de artículos en subasta a partir de una búsqueda por nombre.

Actor: Usuario y Usuario Registrado

Precondiciones: El Usuario Registrado ha sido autenticado en el sistema.

**Postcondiciones:** El usuario ha consultado la lista de Artículos solicitada por búsqueda por nombre.

#### Flujo:

- 1.- El Usuario solicita la lista de artículos subastados introduciendo un nombre.
- 2.- El Sistema devuelve la información de los artículos subastados en el caso que existan datos.
- 3.- El Sistema no devuelve información de artículos subastados con ese nombre de búsqueda.
- 4.- El Usuario consulta la lista de Artículos en subasta en el caso que existan datos.

## Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU2.2

## Nombre: Buscar Artículos Por Categoría

**Descripción:** Permite listar un conjunto de artículos en subasta a partir de una búsqueda por categoría.

Actor: Usuario y Usuario Registrado

**Precondiciones:** El Usuario Registrado ha sido autenticado en el sistema.

**Postcondiciones:** El usuario ha consultado la lista de Artículos solicitada por búsqueda por categoría.

#### Flujo:

- 1.- El Usuario solicita la lista de artículos subastados introduciendo una categoría.
- 2.- El Sistema devuelve la información de los artículos subastados en el caso que existan datos.
- 3.- El Sistema no devuelve información de artículos subastados con esa categoría de búsqueda.
- 4.- El Usuario consulta la lista de Artículos en subasta en el caso que existan datos.

# Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU2.3

#### Nombre: Buscar Artículos Finalizan Hoy

**Descripción:** Permite listar un conjunto de artículos en subasta a partir de una búsqueda por las subastas que finalizan en el día de hoy.

Actor: Usuario y Usuario Registrado

Precondiciones: El Usuario Registrado ha sido autenticado en el sistema.

**Postcondiciones:** El usuario ha consultado la lista de Artículos solicitada con la fecha de subasta que finaliza en el día de hoy.

## Flujo:

- 1.- El Usuario solicita la lista de artículos subastados, que la fecha de fin de la subasta es igual al día de hoy.
- 2.- El Sistema devuelve la información de los artículos subastados en el caso que existan datos.
- 3.- El Sistema no devuelve información de artículos subastados con esa fecha de fin de subasta.
- 4.- El Usuario consulta la lista de Artículos en subasta en el caso que existan datos.

#### Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

#### Código: CU3

# Nombre: Ver Detalle Artículo

Descripción: Permite ver el detalle de un artículo en subasta.

Actor: Usuario y Usuario Registrado

**Precondiciones:** El Usuario Registrado ha sido autenticado en el sistema.

Postcondiciones: El usuario ha consultado el detalle del artículo solicitado.

#### Flujo:

- 1.- El Usuario solicita el detalle de un artículo subastado.
- 2.- El Sistema devuelve la información del artículo subastado. (Nombre, descripción, imagen del artículo, precio puja actual, etc....)
- 3.- El Usuario consulta el detalle del artículo en subasta.

# Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU3.1

# **Nombre: Ver Comentario**

Descripción: Permite ver los comentarios de un vendedor de un artículo en subasta.

Actor: Usuario y Usuario Registrado

Precondiciones: El Usuario Registrado ha sido autenticado en el sistema.

**Postcondiciones:** El usuario ha consultado los comentarios del vendedor del artículo solicitado.

# Flujo:

- 1.- El Usuario solicita los comentarios del vendedor de un artículo subastado.
- 2.- El Sistema devuelve la información de los comentarios del vendedor. (Total positivos, Total neutro, Total negativo).
- 3.- El Usuario consulta los comentarios del vendedor del artículo en subasta.

# Flujo Alternativo:

# 5.4.2. Casos de Uso del Usuario Registrado

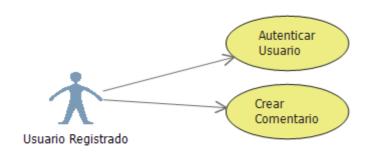


Ilustración 12

# Código: CU4

**Nombre: Autenticar Usuario** 

**Descripción:** Permite validar al usuario en el sistema.

Actor: Usuario Registrado

**Precondiciones:** El usuario ha sido registrado en el sistema.

**Postcondiciones:** El usuario ha sido validado en el sistema y tiene acceso a las funcionalidades pertinentes.

# Flujo:

- 1.- El Sistema solicita los datos de validación al Usuario: NickName y Contraseña.
- 2.- El Usuario introduce los datos requeridos.
- 3.- El Sistema valida el Usuario mediante el nombre y la contraseña facilitados.
- 4.- El Sistema contesta con la validación incorrecta del Usuario.
- 5.- El Sistema contesta con la validación correcta del Usuario.

# Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

#### 5.4.3. Casos de Uso del Comprador

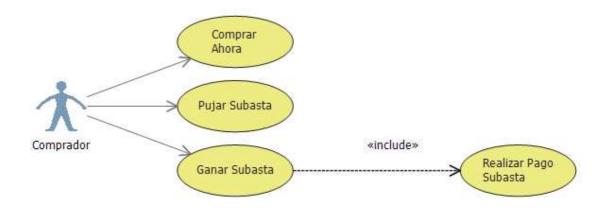


Ilustración 13

#### Código: CU5

Nombre: Pujar Subasta

**Descripción:** Permite realizar una puja sobre una subasta del sistema.

Actor: Comprador

**Precondiciones:** El Comprador ha sido autenticado en el sistema.

Postcondiciones: El usuario ha realizado una puja sobre un artículo subastado en el sistema.

#### Flujo:

- 1.- El Usuario solicita el detalle de un artículo subastado, introduce un precio de puja en el sistema y realiza la acción de pujar.
- 2.- El Sistema comprueba que la puja es válida, es superior a la anterior, la subasta está en estado activa y realiza el registro en el sistema de la puja.
- 3.- El Sistema devuelve al Usuario que la puja se ha realizado correctamente.
- 4.- El Sistema devuelve al Usuario que no se ha podido realizar la puja por el motivo pertinente.
- 5.- El Usuario visualiza su puja si se ha registrado correctamente o visualiza el mensaje de error del sistema.

# Flujo Alternativo:

El usuario cancela la entrada de datos en el sistema.

# Código: CU6

#### Nombre: Comprar Ahora

**Descripción:** Permite realizar una compra de una subasta por el precio establecido por el vendedor.

Actor: Comprador

Precondiciones: El Comprador ha sido autenticado en el sistema.

Postcondiciones: El usuario ha realizado una compra de un artículo.

#### Flujo:

- 1.- El Usuario solicita el detalle de un artículo subastado y realiza la acción de comprar ahora por el precio facilitado por el vendedor.
- 2.- El Sistema comprueba que la compra es válida, y realiza el registro en el sistema de la

#### compra.

- 3.- El Sistema cierra la subasta. El estado de la subasta pasa a Cerrada.
- 4.- El Sistema devuelve al Usuario que ha realizado la compra satisfactoriamente.
- 5.- El Usuario visualiza su compra realizada y la subasta cerrada.

#### Flujo Alternativo:

#### Código: CU7

#### Nombre: Realizar Pago

**Descripción:** Permite realizar el pago de una compra de una subasta ganada.

Actor: Comprador

**Precondiciones:** El Comprador ha sido autenticado en el sistema y ha ganado o comprado un artículo en subasta.

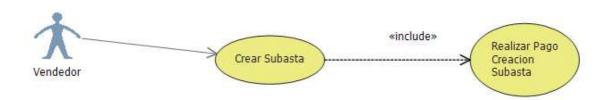
Postcondiciones: El usuario ha realizado el pago de una subasta de un artículo.

#### Flujo:

- 1.- El Usuario es informado de que ha ganado una subasta.
- 2-. El Usuario solicita el detalle de un artículo subastado ganado y realiza la acción de realizar pago.
- 3.- El Sistema comprueba que los datos de la subasta, precio y datos de pago. Realiza la petición de cobro sobre la tarjeta de crédito del comprador.
- 4.- El Sistema recibe la confirmación del pago satisfactorio o no.
- 5.- El Sistema devuelve al Usuario que ha realizado el pago satisfactoriamente o no.
- 6.- El Usuario recibe la información de su pago.

#### Flujo Alternativo:

#### 5.4.4.Casos de Uso del Vendedor



# Ilustración 14

# Código: CU8

Nombre: Subastar Artículo

**Descripción:** Permite subastar un artículo en el Sistema.

Actor: Vendedor

Precondiciones: El Vendedor ha sido autenticado en el sistema.

Postcondiciones: El usuario ha creado una nueva subasta de un artículo.

#### Flujo:

1.- El Usuario introduce los datos del artículo que quiere subastar.( Nombre, Descripción, Imagen, etc...)

- 2.- El Sistema comprueba que los datos del artículo introducido por el vendedor son correctos.
- 3.- El Sistema registra los datos de la subasta y pone el estado Pendiente.
- 4.- El Sistema devuelve al Usuario que ha realizado el registro del artículo correctamente y que debe realizar el pago para poner activa la subasta.
- 5.- El Usuario recibe la información del Sistema conforme los datos introducidos son correctos y queda pendiente el pago de la subasta.

# Flujo Alternativo:

#### Código: CU8.1

#### Nombre: Realizar Pago por Subastar Artículo

**Descripción:** Permite realizar el pago por subastar un artículo en el Sistema.

**Actor:** Vendedor

**Precondiciones:** El Vendedor ha sido autenticado en el sistema y ha puesto un artículo en subasta en el Sistema.

Postcondiciones: El usuario ha pagado la subasta y se activa la nueva subasta del artículo.

#### Flujo:

- 1.- El Usuario es informado de que se han registrado los datos de su nueva subasta.
- 2-. El Usuario realiza la acción de realizar pago.
- 3.- El Sistema comprueba los datos de pago. Realiza la petición de cobro sobre la tarjeta de crédito del vendedor.
- 4.- El Sistema recibe la confirmación del pago satisfactorio o no.
- 5.- El Sistema devuelve al Usuario que ha realizado el pago satisfactoriamente o no.
- 6.- El Sistema cambia el estado de la Subasta de Pendiente a Activa si el pago se confirma como satisfactorio.
- 7.- El Usuario recibe la información de su pago.

# Flujo Alternativo:

# Código: CU9

#### **Nombre: Comentar Subasta**

**Descripción:** Permite realizar comentario a una subasta realizada, en la que ha actuado com comprador o como vendedor.

Actor: Vendedor, Comprador

**Precondiciones:** El Vendedor o Comprador ha sido autenticado en el sistema. Ha finalizado la subasta de un artículo que ha vendido en el caso del Vendedor o que ha comprado en el caso del Comprador.

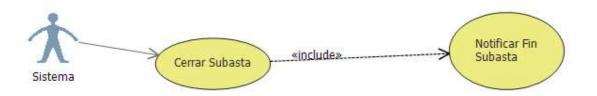
Postcondiciones: El usuario ha creado un comentario sobre una subasta realizada.

# Flujo:

- 1.- El Usuario solicita el detalle de un artículo subastado y realiza la acción de añadir comentario. Puede Elegir entre los tipos Positivo, Neutro o Negativo.
- 2.- El Sistema registra el tipo de comentario del Usuario en el Sistema.

#### Flujo Alternativo:

#### 5.4.5.Casos de Uso del Sistema



# Ilustración 15

# Código: CU10

# Nombre: Cerrar Subasta

**Descripción:** Permite cerrar una subasta en el momento que se ha llegado a la fecha de finalización de esta.

**Actor:** Sistema

Precondiciones: La subasta ha llegado a la fecha de finalización.

Postcondiciones: El sistema ha finalizado la subasta.

#### Flujo:

- 1.- El Sistema finaliza una subasta ya que la fecha de finalización ha sido superada.
- 2.- El Sistema registra la subasta con el estado Cerrada.
- 3.- El Sistema notifica al Vendedor que ha finalizado la subasta vía Email.
- 4.- El Sistema notifica al Comprador en el caso que exista y haya habido alguna puja, que ha ganado la subasta vía Email.

# Flujo Alternativo:

# **CAPÍTULO** 6

# 6 DISEÑO

# 6.1. Diseño Arquitectura Sistema

# 6.1.1. Arquitectura Actual del Sistema

La arquitectura actual del sistema es en tres capas: la capa de presentación, la capa de negocio y la capa de persistencia.

- La Capa de Presentación: Esta capa realiza la presentación al usuario, mostrando todos los componentes que comunican información y capturan la información del usuario. También es conocida como interfaz gráfica. Se comunica con la Capa de Negocio.
- La Capa de Negocio: Esta capa incorpora los componentes que implementan la lógica del negocio en el sistema. Incluye las entidades de negocio que representan los objetos específicos para el dominio del negocio y establecen todas las reglas del negocio. Se comunica con la Capa de Presentación facilitando los datos y recibiendo los datos introducidos por el usuario.
- La Capa de Datos: es la capa que guarda los datos. Desde esta capa se acceden a los datos que está formada por un gestor de base de datos que se encarga del almacenamiento de estos. Se comunica con la Capa de Negocio facilitando los datos y recibiendo.

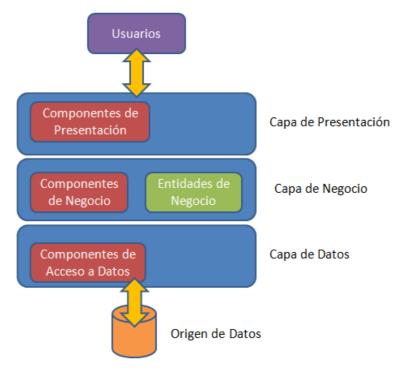


Ilustración 16

#### 6.1.2. Arquitectura del Nuevo Sistema

La nueva arquitectura es muy similar a la de una aplicación de 3 capas, con la diferencia que el Servicio lógicamente no tiene capa de presentación, en su lugar tiene la Capa de Interfaz del Servicio. Aunque se puede disponer del sistema para que se muestre la capa de presentación, en este caso seguirá siendo una arquitectura de 4 capas.

La arquitectura es la siguiente, siendo iguales las Capa de Negocio y Capa de Datos al modelo anterior.

 La Capa de Interfaz del Servicio: es la capa que define las operaciones que provee el servicio. Los consumidores interactuaran con esta capa. Esta capa se comunica con la Capa de Negocio donde traslada la información recibida y recupera los datos pertinentes.

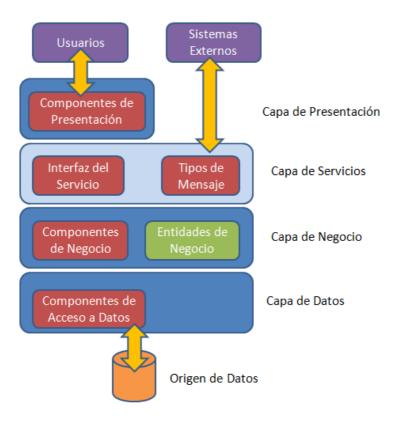


Ilustración 17

El Servicio distribuido deberá proporcionar un interfaz con poca granularidad, es decir, realizará el máximo de operaciones dentro de un método para que el número de llamadas remotas sea el mínimo posible por parte del cliente consumidor. En el diagrama de la interfaz del Servicio se pueden ver los métodos.

El diseño de la capa del servicio es el más importante ya que es donde están definidos los contratos de datos y mensajes que representan los métodos permitidos por el propio servicio. Esta capa se comunicará con la capa de negocio o dominio, donde se interactuaran con los objetos.

En el diseño de la arquitectura de Servicios Web se definirán los fallos o mensajes de error, que pueden surgir para controlar todas las excepciones y comunicar debidamente al cliente.

Se implementan dos soluciones de diseño para el problema, una primera solución con la tecnología de Servicios Web consumidos por ASP.NET y una segunda solución con la tecnología de WCF (Windows Communication Foundation) consumidos por ASP.NET. Cada uno de los diseño producirá como un resultado un prototipo distinto. El ABC (Address, Binding, Contract) del prototipo WCF, será igual al del Servicio Web.

La primera solución de Servicio Web será la siguiente:

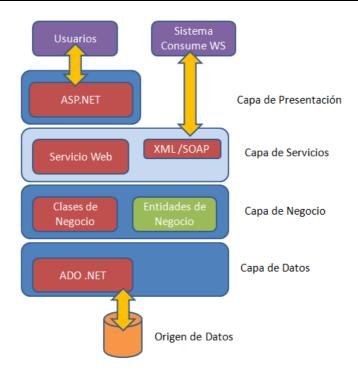


Ilustración 18

La segunda solución de WCF será la siguiente:

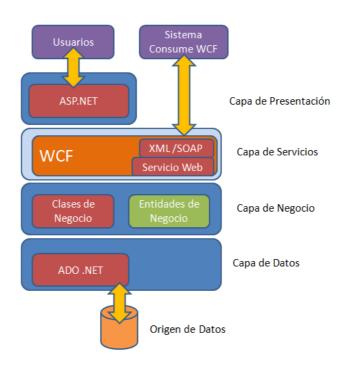


Ilustración 19

El diseño de la arquitectura será distinto para los dos prototipos, no obstante, el resto del diseño así como el análisis previo efectuado es el mismo para los dos prototipos.

En el siguiente diagrama se muestra la arquitectura de la solución final, con la capa de presentación realizada para consumir los servicios creados en los dos prototipos.

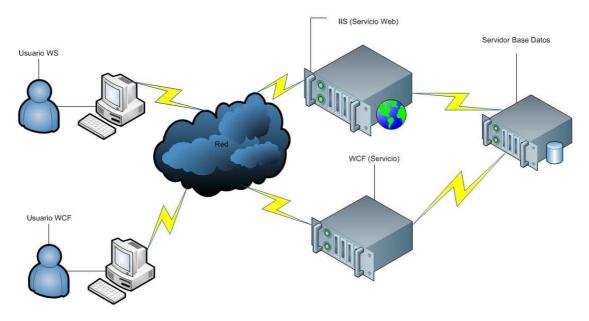


Ilustración 20

# 6.2. Diagrama de Clases

#### 6.2.1.Entidades

En la siguiente figura se puede observar la una clase base TBaseBBDD, que ofrece los métodos necesarios para el acceso a datos. Todas las clases implementandas heredan de esta clase las propiedades y métodos. A su vez, implementan el interface TiBBDD para que el nombre de los métodos sean comunes y reduzca la complejidad en el manejo de las clases.

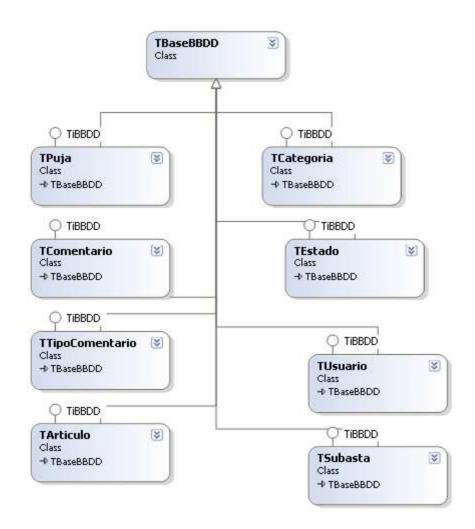


Ilustración 21

En la siguiente figura se muestra la interface utilizada y la clase TDB que se encargará de la conexión a la base de datos.

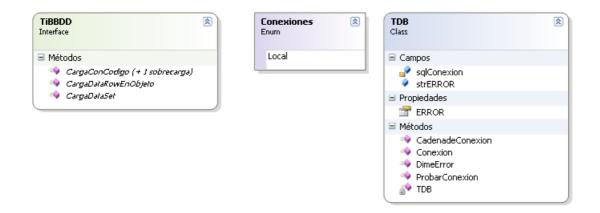


Ilustración 22

En la siguiente figura se muestra la clase TBaseBBDD con todas sus propiedades y métodos.

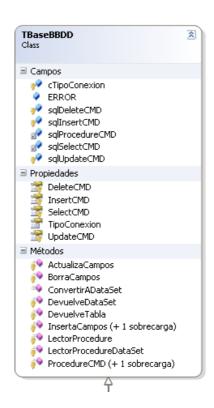


Ilustración 23

La siguiente figura muestra las clases TComentario y TTipoComentario.

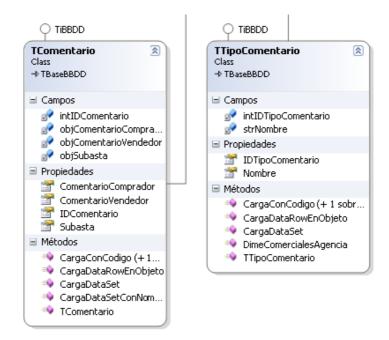


Ilustración 24

La siguiente figura muestra las clases TSubasta y TPuja.

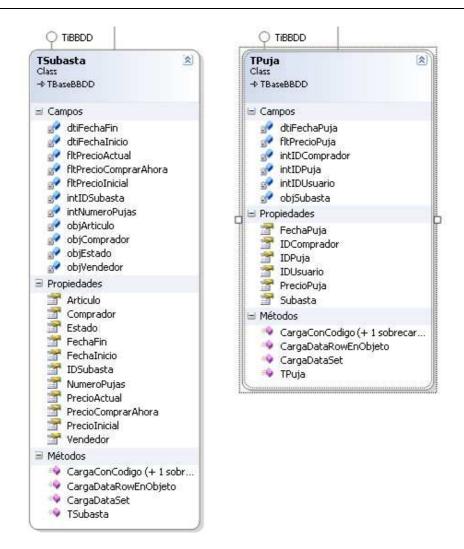


Ilustración 25

La siguiente figura muestra las clases TArticulo y TCategoria.

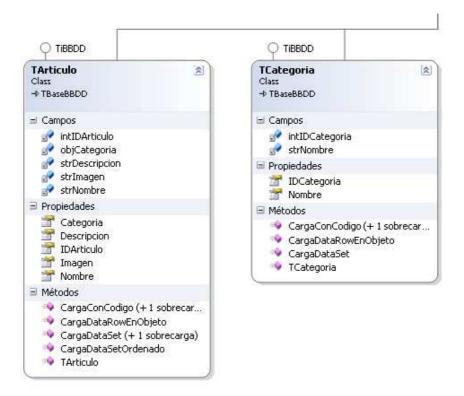


Ilustración 26

La siguiente figura muestra la clase TEstado.

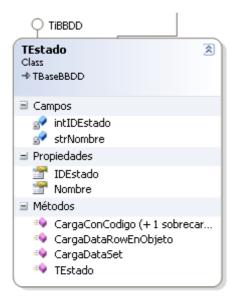


Ilustración 27

La siguiente figura muestra la clase TUsuario.

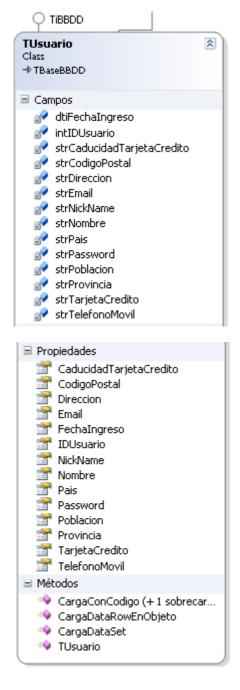


Ilustración 28

#### 6.2.2.Interfaces

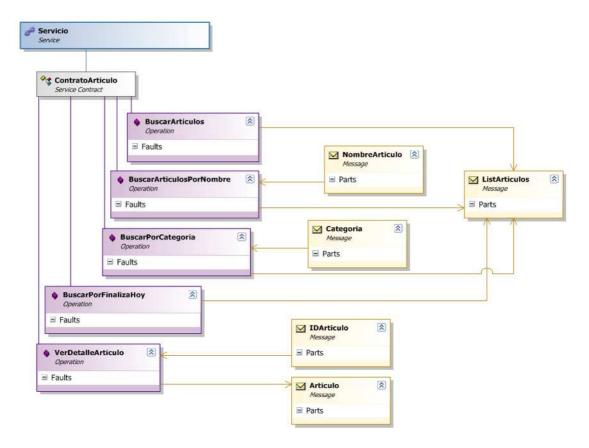


Ilustración 29

# Método: BuscarArticulos

Mensaje Entrada: Sin parámetros.

**Mensaje Salida:** ListArticulos (Listado con todos los artículos de la búsqueda)

# **Método :** BuscarArticulosPorNombre

**Mensaje Entrada:** NombreArticulo (Nombre del artículo que se quiere buscar o parte del nombre).

Mensaje Salida: ListArticulos (Listado con todos los objetos artículos de la búsqueda)

## **Método**: BuscarArticulosPorCategoria

**Mensaje Entrada:** Categoria (Selección de la categoría que se quiere filtrar).

**Mensaje Salida:** ListArticulos (Listado con todos los objetos artículos de la búsqueda)

# Método: BuscarArticulosPorFinalizaHoy

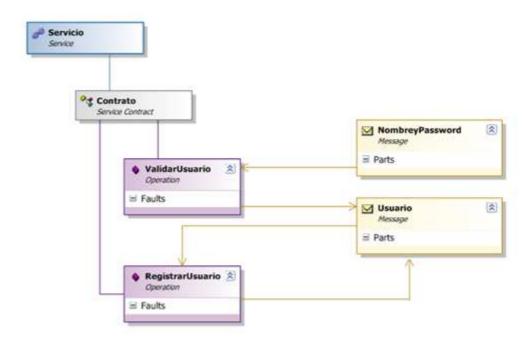
Mensaje Entrada: Sin parámetros.

Mensaje Salida: ListArticulos (Listado con todos los objetos artículos de la búsqueda, todos los artículos que finalicen su subasta en el día de hoy.)

# Método: VerDetalleArticulo

**Mensaje Entrada:** IDArticulo (Identificador del artículo seleccionado para ver su detalle).

**Mensaje Salida:** Articulo (Objeto Articulo que contendrá todos los datos de su detalle.)



#### Ilustración 30

# Método: Validar Usuario

**Mensaje Entrada:** Nombre (cadena con el nickname del usuario que se quiere validar).

Password (cadena con la contraseña del usuario).

Mensaje Salida: Usuario (Objeto Usuario que contendrá todos los datos del usuario validado.)

#### Método: Registrar Usuario

**Mensaje Entrada:** Usuario (Objeto Usuario que contendrá todos los datos del usuario para registrar.)

**Mensaje Salida:** Boolean (Resultado de la operación de registrar a un Usuario)

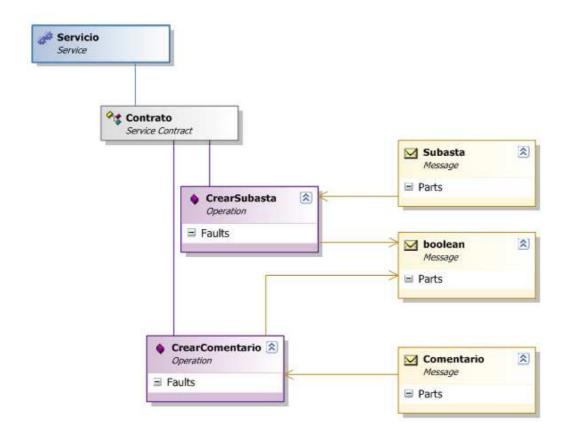


Ilustración 31

# Método: CrearSubasta

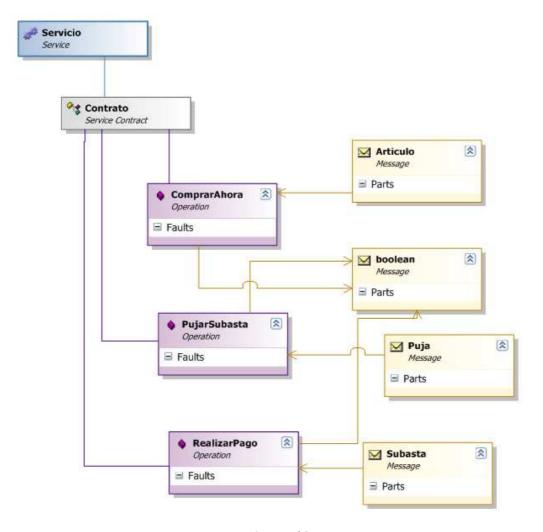
**Mensaje Entrada:** Subasta (Objeto Subasta que contendrá todos los datos de la subasta para crear).

**Mensaje Salida:** Boolean (Resultado de la operación de crear una Subasta)

# **Método**: CrearComentario

Mensaje Entrada: Comentario (Objeto Comentario que contendrá el dato del comentario).

**Mensaje Salida:** Boolean (Resultado de la operación de crear un Comentario)



#### Ilustración 32

# Método: ComprarAhora

Mensaje Entrada: Articulo (Objeto Articulo que contendrá el dato del artículo que se quiere comprar).

Mensaje Salida: Boolean (Resultado de la operación de comprar un artículo)

# Método: PujarSubasta

**Mensaje Entrada:** Puja (Objeto Puja que contendrá el dato de la puja, es decir, el precio y el identificador del artículo por el cual se puja).

**Mensaje Salida:** Boolean (Resultado de la operación de pujar sobre un artículo)

# Método: Realizar Pago

**Mensaje Entrada:** Subasta (Objeto Subasta que contendrá el dato de la subasta ganada en la cual se va a proceder a su pago).

Mensaje Salida: Boolean (Resultado de la operación de pagar una subasta ganada).

# 6.3. Diagrama de Actividad

El siguiente diagrama muestra la visualización de los artículos por los usuarios registrados y no registrados. Para los usuarios registrados se muestran la creación de pujas en las subastas.

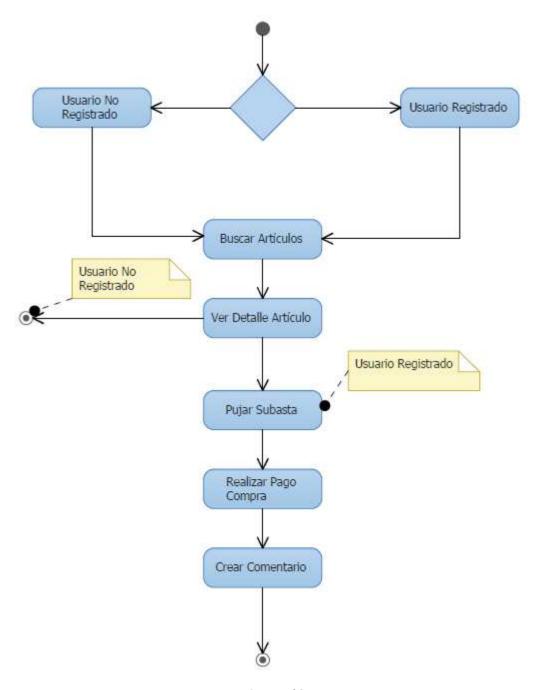


Ilustración 33

El siguiente diagrama muestra la creación de una subasta por el usuario registrado.



Ilustración 34

El siguiente diagrama muestra la compra de un artículo por medio de comprar ahora.



Ilustración 35

Oscar Prol Bravo

# 6.4. Diseño de la Base de Datos

# 6.4.1. Modelo Relacional

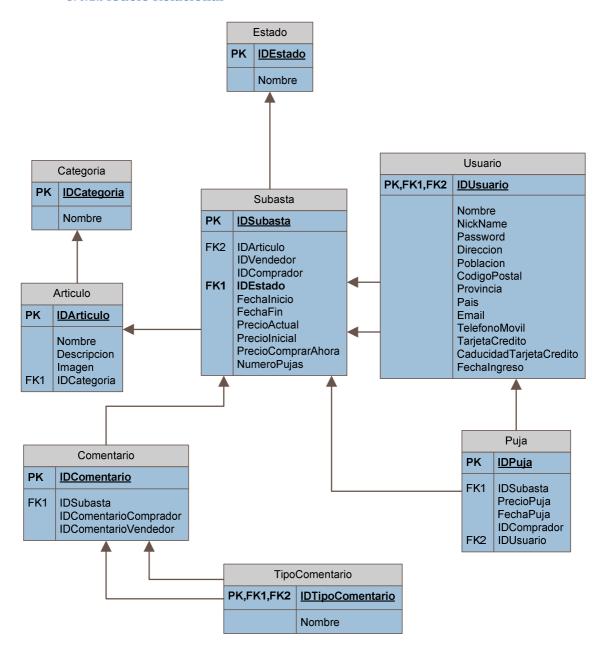


Ilustración 36

# 6.5. Diseño Interfaz Gráfica

# 6.5.1.Interfaz Autenticar Usuario



Ilustración 37

# 6.5.2.Interfaz Registrar Usuario

Da	itos de Registro del U	Isuario	
Nombre :	Escriba texto	Email :	Escriba texto
NickName:	Escriba texto	Teléfono :	Escriba texto
Password:	Escriba texto	Tarjeta de crédito :	Escriba texto
Dirección :	Escriba texto	Caducidad :	Escriba texto
Población :	Escriba texto		
Código Postal :	Escriba texto		
País:	Escriba texto		

Ilustración 38

# 6.5.3.Interfaz Listado Artículos

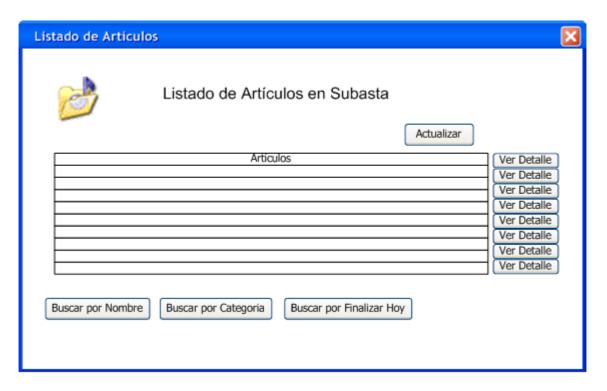


Ilustración 39

# 6.5.4.Interfaz Detalle Artículo



Ilustración 40

# 6.5.5.Interfaz Confirmar Puja



Ilustración 41

# 6.5.6.Interfaz Crear Subasta Artículo

<b>₽</b>	itos de Registro del Artí	culo de la Subasta
Artículo :	Escriba texto	
Nombre :	Escriba texto	
Descripción :	Escriba texto	
Imagen :	Escriba texto	
Fecha Inicio:	Escriba texto	
Fecha Fin:	Escriba texto	Precio Comprar Ahor
Precio Inicial:	Escriba texto	Escriba texto

Ilustración 42

# 6.5.1.Interfaz Comentarios



Ilustración 43

# 7 Manual de Instalación

# 7.1. Requisitos Previos

#### **7.1.1.Sistema**

El desarrollo del proyecto se ha realizado sobre la plataforma .NET de Microsoft, por lo tanto el debe disponer de un Sistema Operativo Microsoft con el Framework 2.0 y el Framework 3.0. El sistema operativo recomendado es Microsoft, aunque existan distribuciones compatibles del .NET Framework para otros sistemas operativos como Linux, por ejemplo Mono, no han sido probadas en la implementación de este sistema por lo que no se puede garantizar el correcto funcionamiento ya que esta fuera del ámbito del proyecto.

El desarrollo del Web Service se ha realizado sobre la plataforma Framework .NET 2.0, mientras que el desarrollo del WCF (Windows Comunication Foundation) se ha realizado sobre la plataforma Framework .NET 3.0.

La implementación de los clientes para tanto el Web Service como el WCF se han implementado por medio de ASP.NET, ya que son unos sitios web que consumen los datos mediante estas tecnologías.

Para el acceso a datos por parte del Web Service y el WCF, parte servidora, se ha implementado por medio de ADO.NET. El servidor de Base de Datos utilizado para la implementación es SQL Server 2008, y el acceso a este se realiza por el SQL Server Native Client.

El ambiente de desarrollo ha sido Visual Studio 2010, para poder realizar los dos prototipos.

Las aplicaciones servidora del prototipo 1 se publicará por medio del Internet Information Server, que albergará el Web Service por el lado servidor.

En el lado Cliente también se encontrará esta aplicación para realizar la publicación de los sitios web.

Por otro lado, el servidor WCF se publicará por medio de un Servicio de Windows, ya que el modo de conexión es por TCP y para diferenciar el ambiente y no utilizar de nuevo el IIS, se ha optado por esta solución.

El resumen de los prototipos es la siguiente:

#### Prototipo 1:

Parte Servidora:

SQL Server 2008, SQL Server Native Client

Servidor IIS: Servicio Web (Framework 2.0)

#### Parte Cliente:

Servidor IIS: Sitio Web nº1 (ASP.NET)

# Prototipo 2:

Parte Servidora:

SQL Server 2008, SQL Server Native Client

Servidor IIS: WCF (Framework 3.0)

Parte Cliente:

Servidor IIS: Sitio Web nº2 (ASP.NET)

En los dos prototipos comparten el mismo Servidor de Base de Datos de la parte Servidora. En el caso del Servidor IIS puede ser diferente, no obstante los pasos de la configuración son los mismos para cualquiera de los dos modelos.

En la parte Servidora, el Servidor IIS servirá un sitio web para la disponibilidad las imágenes por el protocolo Http de los artículos. No obstante, este servidor podría estar albergado en otro sitio completamente distinto. Por lo tanto se deberá instalar un sitio web donde publique la carpeta imágenes y que publique por el puerto 80 http, ya que será el sitio de consulta en el dominio www.subatas.es.

En el apartado de código se adjunta una carpeta "sources" donde se encuentran todos los códigos fuentes de los dos prototipos. Así mismo, en la carpeta "Base de datos" se encuentra un backup completo de la base de datos (incluyen datos de test) y los scripts para la creación de un base de datos vacía.

La estructura de los códigos entregados es la siguiente:

- Sources:
  - o Prototipo 1
    - Servidor
    - Cliente
  - o Prototipo 2
    - Servidor
    - Cliente
  - o Base de Datos
    - Scripts

La estructura de los prototipos entregados es la siguiente:

- Publish:

- o Prototipo 1
  - Servidor
  - Cliente
- Prototipo 2
  - Servidor
  - Cliente

La configuración de los servidores de nombre es necesaria para poder mostrar las imágenes y para poder conocer el nombre de servidor de Base de Datos. El fichero de configuración se encuentra en el directorio c:\windows\system32\drivers\etc, con el nombre "hosts", en la siguiente figura se puede visualizar el cambio realizado para poner el servidor "www.subastas.es" y "subastas" en el localhost (127.0.0.1).

```
File Edit Format View Help

# Copyright (c) 1993-2009 Microsoft Corp.

# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.

# This file contains the mappings of IP addresses to host names. Each entry should be kept on an individual line. The IP address should be placed in the first column followed by the corresponding host name.

# The IP address and the host name should be separated by at least one space.

# Additionally, comments (such as these) may be inserted on individual lines or following the machine name denoted by a '#' symbol.

# For example:

# 102.54.94.97 rhino.acme.com # source server 38.25.63.10 x.acme.com # x client host

# localhost name resolution is handled within DNS itself.

# 127.0.0.1 localhost

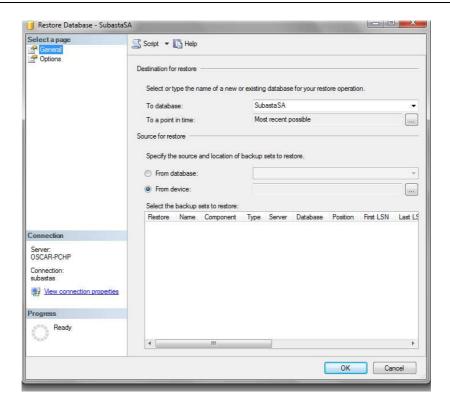
# 127.0.0.1 www.subastas.es

127.0.0.1 subastas.
```

Ilustración 44

#### 7.2. Instalación Base Datos

La restauración de la base de datos es relativamente sencilla. El primer paso es abrir el apartado de base de datos, y con el botón derecho realizar la operación "Restaurar Base de Datos".



# Ilustración 45

El nombre de la base de datos por defecto se debe denominar "SubastaSA", no obstante se puede crear con otro nombre pero habrá que tenerlo en cuenta en la configuración de las conexiones en los prototipos.

El medio de restauración es un fichero desde un dispositivo, este será el fichero facilitado.

Una vez, que se ha seleccionado el fichero correctamente.

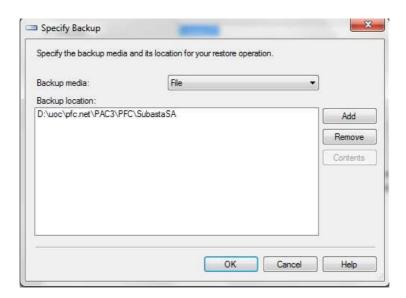


Ilustración 46

Finalmente se selecciona la base de datos y se procede a su restauración.

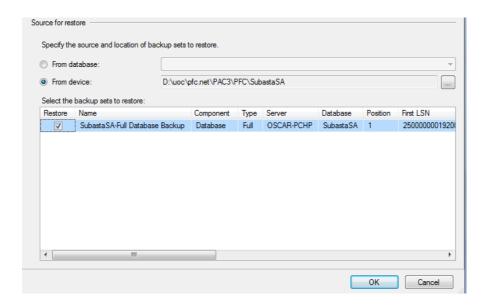


Ilustración 47

Una vez, esta restaurada la base de datos se deberá crear un usuario Subasta, con contraseña Subasta, con permisos de escritura, ejecución de procedimientos y lectura de toda la base de datos que acabos de crear. El nombre y contraseña del usuario, al igual que el nombre de la base de datos pueden ser modificados, teniéndolos en cuenta para la posterior configuración.

Se tiene que crear una tarea programa del servidor de Base de Datos que se ejecute cada minuto para ejecutar el procedimiento almacenado "FinalizaSubasta". Este procedimiento realiza el control de las subastas y las finaliza en el momento que el tiempo de finalización se igual o inferior al actual.

# 7.3. Instalación Prototipo 1 : WS

#### 7.3.1. Aplicación Servidor

La carpeta Prototipo 1 \Servidor se publica por medio del Internet Information Server. Se crea un nuevo directorio virtual para la aplicación cliente WebService.

Particularmente en este sitio web hay que tener en cuenta que publique por el puerto 7876 para tener los parámetros de conexión por defecto. En caso contrario habrá que modificar también el cliente.

La configuración de la página asp.net es la siguiente (se puede verificar la correcta carga desde el IIS):

```
<configuration>
     <appSettings/>
```

Como se puede ver, la configuración del Servicio Web es sencilla.

Hay que tener en cuenta que el servidor de Base de datos está en la IP 127.0.0.1, en el caso que se cambiará habría que modificar este dato, al igual que los datos de conexión como el nombre de la base de datos y los parámetros de seguridad: nombre y contraseña del usuario.

#### 7.3.2. Aplicación Cliente

La carpeta Prototipo 1 \Cliente se publica por medio del Internet Information Server. Se crea un nuevo directorio virtual para la aplicación cliente SubastasWS1.

La configuración de la página asp.net es la siguiente (se puede verificar la correcta carga desde el IIS):

```
<configuration>
       <appSettings>
              <add key="PathImagenes" value="C:\\Sub1\\imagenes\\"/>
              <add key="localhost.Subastas"</pre>
value="http://localhost:7876/Subastas.asmx"/>
       </appSettings>
       <connectionStrings>
       </connectionStrings>
       <system.web>
              <compilation debug="true" targetFramework="4.0">
                     <expressionBuilders>
                            <add expressionPrefix="Themes"</pre>
type="ThemesExpressionBuilder"/>
                     </expressionBuilders>
              </compilation>
              <customErrors mode="RemoteOnly"/>
              <authentication mode="Windows"/>
              <!-- Use any of the following values for the styleSheetTheme
attribute: "Fruits", "Forest", "Literature".-->
              <pages styleSheetTheme="Forest"</pre>
controlRenderingCompatibilityVersion="3.5" clientIDMode="AutoID"/>
       </system.web>
</configuration>
```

En el fichero de configuración de la página web SubastasWS1 únicamente hay que tener en cuenta la configuración del Servicio Web, que según se marca en el fichero es:

```
<add key="localhost.Subastas" value="http://localhost:7876/Subastas.asmx"/>
```

Por lo que el servidor se debe encontrar en el servidor Localhost (127.0.0.1) publicando por HTTP por el puerto 7876.

La página utilizar estilos predefinidos que se pueden configurar desde el Web.config. La línea de configuración es la siguiente:

```
<pages styleSheetTheme="Forest" controlRenderingCompatibilityVersion="3.5"
clientIDMode="AutoID"/>
```

# 7.4. Instalación Prototipo 2: WCF

#### 7.4.1. Aplicación Servidor

La instalación del Servidor de WCF es diferente al Web Services ya que se ha optado por hospedar el Servidor en un Servicio de Windows. Aunque se podría hospedar en el Internet Information Server.

Primero de todo registraremos el Servicio de Windows. Para realizar este paso ejecutaremos desde la carpeta de Servidor del prototipo 2, el siguiente comando: "installutil consoleapplication1.exe", ya que es el nombre que se ha dado al proyecto de consola.

```
Affected parameters are:
logiconsole =
logile = C:\TestWCF\ConsoleApplication1\ConsoleApplication1\bin\Debug\Consol
eApplication1.InstallLog
assemblypath = C:\TestWCF\ConsoleApplication1\ConsoleApplication1\bin\Debug\Consol
onsoleApplication1.exe
Installing service WCFSubastasServicio...
Creating EventLog source WCFSubastasServicio in log Application...
An exception occurred during the Install phase.
System.ComponentModel.Win32Exception: The specified service already exists
The Rollback phase of the installation is beginning.
See the contents of the log file for the C:\TestWCF\ConsoleApplication1\ConsoleApplication1\bin\Debug\ConsoleApplication1\bin\Debug\ConsoleApplication1\bin\Debug\ConsoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\consoleApplication1\bin\Debug\ConsoleApplication1\co
```

Ilustración 48

Una vez se ha instalado correctamente el servicio, se podrá ejecutar en gestor de servicios de Windows, desde la línea de comandos "services.msc". Si hay problemas para registrar el servicio habrá que deshabilitar UAC del usuario mediante el panel de gestión de usuarios de Windows y posteriormente reiniciar el sistema.

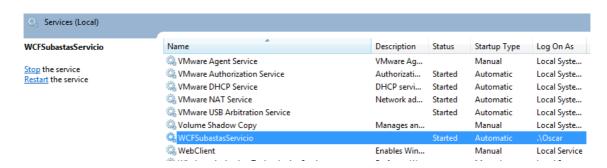


Ilustración 49

El nombre del servicio es WCFSubastasServicio. Con botón derecho se puede arrancar el servicio. Si da problemas al arrancar, hay que validar que la cuenta de arranque del servicio tiene los permisos suficientes, por medio de propiedades del servicio se pueden cambiar los datos de usuario y contraseña para arrancar el servicio.

Una vez arrancado el servicio, el Servidor de WCF se estará ejecutando y escuchando peticiones según se ha configurado en el app.config de la aplicación. En el caso por defecto escuchará peticiones por TCP en el puerto 9090.

La configuración de la aplicación es la siguiente:

Mediante este fichero de configuración "app.config" se podrían modificar fácilmente la forma de conexión, tales como puerto o el protocolo, para configurarlo por http por ejemplo.

#### 7.4.2. Aplicación Cliente

La configuración del cliente es igual que la aplicación cliente del prototipo 1.

La carpeta Prototipo 2 \Cliente se publica por medio del Internet Information Server. Se crea un nuevo directorio virtual para la aplicación cliente SubastasWCF2.

La configuración de la página asp.net es la siguiente (se puede verificar la correcta carga desde el IIS):

```
<?xml version="1.0"?>
<configuration>
       <appSettings>
<add key="PathImagenes" value="C:\\Sub1\\imagenes\\"/>
       </appSettings>
       <connectionStrings>
       </connectionStrings>
       <system.web>
               <compilation debug="true" targetFramework="4.0">
               <expressionBuilders>
               <add expressionPrefix="Themes" type="ThemesExpressionBuilder"/>
               </expressionBuilders>
               </compilation>
               <customErrors mode="RemoteOnly"/>
               <authentication mode="Windows"/>
<pages styleSheetTheme="Literature" controlRenderingCompatibilityVersion="3.5"</pre>
clientIDMode="AutoID"/>
    </system.web> <system.serviceModel> <bindings> <netTcpBing
<binding name="NetTcpBinding_IService1" closeTimeout="00:01:00"</pre>
                                                              <netTcpBinding>
     openTimeout="00:01:00" receiveTimeout="00:10:00" sendTimeout="00:01:00"
     transactionFlow="false" transferMode="Buffered"
transactionProtocol="OleTransactions"
     hostNameComparisonMode="StrongWildcard" listenBacklog="10"
maxBufferPoolSize="524288"
     maxBufferSize="65536" maxConnections="10" maxReceivedMessageSize="65536">
     <readerQuotas maxDepth="32" maxStringContentLength="8192"</pre>
maxArrayLength="16384"
      maxBytesPerRead="4096" maxNameTableCharCount="16384" />
     <reliableSession ordered="true" inactivityTimeout="00:10:00"</pre>
      enabled="false" />
     <security mode="Transport">
    <transport clientCredentialType="Windows" protectionLevel="EncryptAndSign" />
    <message clientCredentialType="Windows" />
     </security>
    </binding>
   </netTcpBinding>
  </bindings>
  <client>
   <endpoint address="net.tcp://localhost:9090/" binding="netTcpBinding"</pre>
    bindingConfiguration="NetTcpBinding_IService1" contract="Service1.IService1"
    name="NetTcpBinding_IService1">
    <identity>
     <userPrincipalName value="Oscar-PCHP\Oscar" />
                   </endpoint> </client> </system.serviceModel>
    </identity>
</configuration>
```

En el fichero de configuración de la página ASP.NET se puede ver la conexión contra el localhost por el puerto 9090 por el protocolo net.tcp. En el caso que se quieran cambiar los parámetros del servidor se deberán cambiar también los del cliente. La línea de configuración es la siguiente:

# 8 Manual de Usuario

# 8.1. Aplicación Cliente

#### 8.1.1.Página Web

Los prototipos han sido desarrollados con la tecnología ASP .NET de Microsoft, por lo tanto se deberán ejecutar las aplicaciones cliente en un navegador. Cualquiera de los navegadores convencionales es compatible con los prototipos.

Los prototipos están diseñador con una plantilla que hace uso de los CSS (Cascading Style Sheets), y de un webSiteMap para poder realizar el menú y mostrar en la página en la que se encuentra actualmente.

Así pues para arrancar cualquiera de los dos prototipos se deberá abrir un navegador web y poner la dirección en la que se encuentre albergado el servidor IIS que acepte las peticiones, ya sea por el puerto 80 (por defecto) o por el puerto correspondiente, el cual deberá ser indicado en el navegador después del nombre o dirección con dos puntos y el número del puerto al que se quiere acceder.

En los siguientes capítulos del documento se detallan cada una de las páginas (pantallas) que ofrecen cada uno de los prototipos.

El detalle de las pantallas corresponde entre los dos prototipos, por lo que este documento explica con el mismo detalle las figuras de los dos prototipos.

En la siguiente tabla se pueden algunos de los usuarios de test que están creados en el sistema:

Usuario	Contraseña
Peter	Peter1234
Juan	Juan1234
Josep	Josep1234
Pedro	Pedro1234

# 8.2. Prototipo 1: Aplicación Cliente: Subastas ws1

La página principal es la primera página que carga el navegador al acceder al servidor web. En la figura número uno se puede ver, esta página tiene como contenido dinámico el menú de la izquierda que facilita el acceso a los diferentes apartados de la página.



Las siguientes páginas corresponden a la presentación de la empresa con contenidos estáticos, la página "Acerca De" y "Contactar", que corresponden a las siguientes figuras dos y tres.

La página "Acerca de la Empresa" muestra un texto estático de presentación.



## Ilustración 51

La página "Contactar" solicita unos datos como el nombre y la dirección de correo para poder subscribirse a las noticias de la empresa, pudiendo seleccionar la categoría de las subastas y el formato del correo.



Ilustración 52

La página "Subasta" es la principal de la aplicación de subastas, desde esta página se puede acceder al detalle de los artículos y se puede validar en el sistema como usuario registrado. En la parte superior tiene un enlace a la página de validación de usuarios.

En la parte central se pueden realizar filtros sobre los artículos que se muestran en la tabla. Los filtros pueden ser por Categoría seleccionando en la parte superior con el desplegable que facilita todos los tipos de categorías que hay en el sistema, o por medio del texto buscar artículo, por medio el cual se pueden filtra nombres de artículos.

En la tabla se muestran los datos del artículo, estos datos son:

- Un identificador de la subasta que se está visualizando.
- El nombre del artículo que se está subastando.
- El Precio actual de la Subasta.
- El Precio "Comprar Ahora" que tiene el respectivo artículo.
- La Fecha en la que finaliza la subasta.



Ilustración 53

En la siguiente figura se puede observa el filtro de categorías desplegado, y donde se pueden seleccionar cada una de ellas.



Ilustración 54

En la siguiente figura se muestra el detalla de la subasta, a este detalle se accede pulsando sobre el nombre del artículo en la subasta.



Ilustración 55

En este detalle de la subasta se muestran los siguientes datos:

- Nombre del Artículo: Es el nombre del artículo que se está subastando.
- <u>Descripción del Artículo:</u> Es la descripción del artículo que se está subastando.
- Fecha de Inicio: Es la fecha de inicio de la subasta.
- Fecha Final: Es la fecha de finalización de la subasta.
- <u>Precio Comprar ahora:</u> Es el precio que tiene la subasta en "Comprar Ahora".
- <u>Precio Actual:</u> Es el precio actual que tiene la subasta.
- <u>Estado de la Subasta:</u> Es el estado en el que se encuentra la subasta, puede ser activo o finalizado.
- <u>Número de Pujas:</u> Es el número de pujas que tiene la subasta en cuestión.
- Vendedor: Es el nombre del vendedor con su correspondiente porcentaje de valoración.
- <u>Comprador:</u> Es el nombre del comprador con su correspondiente porcentaje de valoración.

Desde la página del detalle del artículo se pueden realizar las dos operaciones principales de una subasta:

- <u>Pujar</u>: Mediante el texto solicitado, se introduce la cantidad que se quiere pujar, siempre tendrá que ser superior al precio actual, si no la puja por la subasta no se podrá realizar.
- <u>Comprar Ahora:</u> mediante el botón que se encuentra la parte derecha de la imagen del artículo se puede efectuar la compra por el precio que se detalla.

La siguiente figura muestra la página de validación, a esta se accede por la parte superior en el apartado "Validar". Este acceso está disponible desde cualquier página.

Mediante esta página, un usuario se puede validar en el sistema, para poder realizar las operaciones que se precisen. Las operaciones que son limitadas al usuario validado son: la de realizar una puja de una subasta, efectuar el comprar ahora de una subasta, crear una nueva subasta, y finalmente, realizar un voto a un comprador o vendedor, pero en este caso necesita ser uno de estos actores de la subasta.



## Ilustración 56

Desde la página de Validar el usuario se puede acceder al Registro del usuario, para que los usuarios que no estén registrados en el sistema lo puedan hacer.

La página de registro de usuario en el sistema es la que se muestra en la siguiente figura.



Ilustración 57

Los datos que se solicitan al usuario para el registro son los siguientes:

- <u>Usuario:</u> Es el nombre del usuario en el sistema o apodo.
- Contraseña: Es la contraseña del usuario en el sistema.
- Nombre: Es el nombre real del usuario.
- <u>Dirección:</u> Es la dirección del usuario.

- Población: Es la población del usuario.
- Código Postal: Es el Código Postal del usuario.
- Provincia: Es la provincia del usuario.
- País: Es el país del usuario.
- Email: Es la dirección de correo del usuario.
- Teléfono Móvil: Es el teléfono de contacto del usuario.
- <u>Tarjeta Crédito:</u> Es la tarjeta de crédito del usuario con la que quiere realizar las transacciones.
- <u>Caducidad Tarjeta</u>: Es la fecha de caducidad de la tarjeta con la que quiere realizar las transacciones.

En la siguiente figura se muestra la pantalla de subastas, pero con la diferencia que hay un usuario validado en el sistema, en el caso de la figura siguiente es "Peter". Al lado del nombre del usuario aparece el acceso directo a "Mis Subastas", para poder visualizar las subastas que han finalizado y que actúa como vendedor o comprador de esta.



## Ilustración 58

En la siguiente figura se muestra la página de creación de una subasta, a la cual se accede desde el botón de "Subastar Artículo" desde la página de subastas.



### Ilustración 59

En la página de subastar artículo se solicitan los siguientes datos:

- <u>Categoría:</u> es la categoría donde se quiere clasificar el artículo.

- Nombre del Artículo: Es el nombre del artículo que se está subastando.
- Descripción del Artículo: Es la descripción del artículo que se está subastando.
- Imagen del Artículo: Es la fotografía del artículo que se está subastando.
- Fecha Final: Es la fecha de finalización de la subasta.
- <u>Precio Comprar ahora:</u> Es el precio que tiene la subasta en "Comprar Ahora".
- <u>Precio Inicial</u>: Es el precio inicial con el que se quiere partir en la subasta.

En la siguiente figura se muestra "Mis Subastas" que es una tabla de las subastas en las que el usuario validado a participado como Comprador o como Vendedor.



#### Ilustración 60

En la siguiente figura se puede ver el detalle de un artículo con la diferencia que la subasta está finalizada, por lo tanto únicamente es accesible al usuario Comprador o al usuario Vendedor, para poder acceder al acceso directo de "Votar", en este caso como el usuario es "Peter", y actuó como Comprador, puede realizar el voto de los comentarios sobre el Vendedor que es el usuario "Juan".



#### Ilustración 61

En la siguiente figura se muestra la pantalla de votación, en la que se crea un comentario sobre el Comprador o Vendedor de la subasta. Los comentarios pueden ser de tres tipos: Positivo,

Neutro o Negativo. Esta votación es la que se tiene en cuenta para el cálculo del porcentaje de los usuarios, que es el número de votos positivos entre el número de votos totales.



### Ilustración 62

En la siguiente pantalla se puede visualizar la votación sobre un determinado usuario que está actuando en una subasta como comprador o como vendedor.



## Ilustración 63

La siguiente figura muestra la pantalla de registro de pujas de un artículo, a esta se accede mediante el detalle de un artículo. Se pueden ver únicamente tres datos de cada puja, estos datos son los siguientes:

- Nombre Usuario: Nombre del usuario (apodo) que realiza la puja.
- <u>Precio Puja:</u> Precio de la puja.
- Fecha Puja: Fecha en la que se realizo la puja.



Ilustración 64

## 8.3. Prototipo 2: Aplicación Cliente: Subastas wcf2

La página principal es la primera página que carga el navegador al acceder al servidor web. En la figura número uno se puede ver, esta página tiene como contenido dinámico el menú de la izquierda que facilita el acceso a los diferentes apartados de la página.



## Ilustración 65

Las siguientes páginas corresponden a la presentación de la empresa con contenidos estáticos, la página "Acerca De" y "Contactar", que corresponden a las siguientes figuras dos y tres.

La página "Acerca de la Empresa" muestra un texto estático de presentación.



#### Ilustración 66

La página "Contactar" solicita unos datos como el nombre y la dirección de correo para poder subscribirse a las noticias de la empresa, pudiendo seleccionar la categoría de las subastas y el formato del correo.



La página "Subasta" es la principal de la aplicación de subastas, desde esta página se puede acceder al detalle de los artículos y se puede validar en el sistema como usuario registrado. En la parte superior tiene un enlace a la página de validación de usuarios.

En la parte central se pueden realizar filtros sobre los artículos que se muestran en la tabla. Los filtros pueden ser por Categoría seleccionando en la parte superior con el desplegable que facilita todos los tipos de categorías que hay en el sistema, o por medio del texto buscar artículo, por medio el cual se pueden filtra nombres de artículos.

En la tabla se muestran los datos del artículo, estos datos son:

- Un identificador de la subasta que se está visualizando.
- El nombre del artículo que se está subastando.
- El Precio actual de la Subasta.
- El Precio "Comprar Ahora" que tiene el respectivo artículo.
- La Fecha en la que finaliza la subasta.



Ilustración 68

En la siguiente figura se puede observa el filtro de categorías desplegado, y donde se pueden seleccionar cada una de ellas.



En la siguiente figura se muestra el detalla de la subasta, a este detalle se accede pulsando sobre el nombre del artículo en la subasta.



## Ilustración 70

En este detalle de la subasta se muestran los siguientes datos:

- Nombre del Artículo: Es el nombre del artículo que se está subastando.
- <u>Descripción del Artículo:</u> Es la descripción del artículo que se está subastando.
- Fecha de Inicio: Es la fecha de inicio de la subasta.
- <u>Fecha Final:</u> Es la fecha de finalización de la subasta.
- Precio Comprar ahora: Es el precio que tiene la subasta en "Comprar Ahora".
- <u>Precio Actual:</u> Es el precio actual que tiene la subasta.
- <u>Estado de la Subasta:</u> Es el estado en el que se encuentra la subasta, puede ser activo o finalizado.
- <u>Número de Pujas:</u> Es el número de pujas que tiene la subasta en cuestión.
- Vendedor: Es el nombre del vendedor con su correspondiente porcentaje de valoración.

- <u>Comprador:</u> Es el nombre del comprador con su correspondiente porcentaje de valoración.

Desde la página del detalle del artículo se pueden realizar las dos operaciones principales de una subasta:

- <u>Pujar</u>: Mediante el texto solicitado, se introduce la cantidad que se quiere pujar, siempre tendrá que ser superior al precio actual, si no la puja por la subasta no se podrá realizar.
- <u>Comprar Ahora:</u> mediante el botón que se encuentra la parte derecha de la imagen del artículo se puede efectuar la compra por el precio que se detalla.

La siguiente figura muestra la página de validación, a esta se accede por la parte superior en el apartado "Validar". Este acceso está disponible desde cualquier página.

Mediante esta página, un usuario se puede validar en el sistema, para poder realizar las operaciones que se precisen. Las operaciones que son limitadas al usuario validado son: la de realizar una puja de una subasta, efectuar el comprar ahora de una subasta, crear una nueva subasta, y finalmente, realizar un voto a un comprador o vendedor, pero en este caso necesita ser uno de estos actores de la subasta.



Ilustración 71

Desde la página de Validar el usuario se puede acceder al Registro del usuario, para que los usuarios que no estén registrados en el sistema lo puedan hacer.

La página de registro de usuario en el sistema es la que se muestra en la siguiente figura.



Ilustración 72

Los datos que se solicitan al usuario para el registro son los siguientes:

- <u>Usuario</u>: Es el nombre del usuario en el sistema o apodo.
- Contraseña: Es la contraseña del usuario en el sistema.
- Nombre: Es el nombre real del usuario.
- Dirección: Es la dirección del usuario.
- Población: Es la población del usuario.
- <u>Código Postal:</u> Es el Código Postal del usuario.
- Provincia: Es la provincia del usuario.
- País: Es el país del usuario.
- Email: Es la dirección de correo del usuario.
- Teléfono Móvil: Es el teléfono de contacto del usuario.
- <u>Tarjeta Crédito:</u> Es la tarjeta de crédito del usuario con la que quiere realizar las transacciones.
- <u>Caducidad Tarjeta:</u> Es la fecha de caducidad de la tarjeta con la que quiere realizar las transacciones.

En la siguiente figura se muestra la pantalla de subastas, pero con la diferencia que hay un usuario validado en el sistema, en el caso de la figura siguiente es "Peter". Al lado del nombre del usuario aparece el acceso directo a "Mis Subastas", para poder visualizar las subastas que han finalizado y que actúa como vendedor o comprador de esta.



Ilustración 73

En la siguiente figura se muestra la página de creación de una subasta, a la cual se accede desde el botón de "Subastar Artículo" desde la página de subastas.



Ilustración 74

En la página de subastar artículo se solicitan los siguientes datos:

- <u>Categoría:</u> es la categoría donde se quiere clasificar el artículo.
- Nombre del Artículo: Es el nombre del artículo que se está subastando.
- <u>Descripción del Artículo:</u> Es la descripción del artículo que se está subastando.
- Imagen del Artículo: Es la fotografía del artículo que se está subastando.
- <u>Fecha Final:</u> Es la fecha de finalización de la subasta.
- <u>Precio Comprar ahora:</u> Es el precio que tiene la subasta en "Comprar Ahora".
- Precio Inicial: Es el precio inicial con el que se quiere partir en la subasta.

En la siguiente figura se muestra "Mis Subastas" que es una tabla de las subastas en las que el usuario validado a participado como Comprador o como Vendedor.



Ilustración 75

En la siguiente figura se puede ver el detalle de un artículo con la diferencia que la subasta está finalizada, por lo tanto únicamente es accesible al usuario Comprador o al usuario Vendedor, para poder acceder al acceso directo de "Votar", en este caso como el usuario es "Peter", y actuó como Comprador, puede realizar el voto de los comentarios sobre el Vendedor que es el usuario "Juan".



## Ilustración 76

En la siguiente figura se muestra la pantalla de votación, en la que se crea un comentario sobre el Comprador o Vendedor de la subasta. Los comentarios pueden ser de tres tipos: Positivo, Neutro o Negativo. Esta votación es la que se tiene en cuenta para el cálculo del porcentaje de los usuarios, que es el número de votos positivos entre el número de votos totales.



## Ilustración 77

En la siguiente pantalla se puede visualizar la votación sobre un determinado usuario que está actuando en una subasta como comprador o como vendedor.



La siguiente figura muestra la pantalla de registro de pujas de un artículo, a esta se accede mediante el detalle de un artículo. Se pueden ver únicamente tres datos de cada puja, estos datos son los siguientes:

- Nombre Usuario: Nombre del usuario (apodo) que realiza la puja.
- Precio Puja: Precio de la puja.
- <u>Fecha Puja:</u> Fecha en la que se realizo la puja.



Ilustración 79

# 9 Objetivos Conseguidos en el Proyecto

El objetivo principal de este proyecto está establecido en la parte inicial de esta memoria. En resumen, se puede reducir a dos conceptos principales: el análisis comparativo entre la arquitectura de Windows Comunication Foundation y los Servicios Web XML, y por otro lado, la implementación de estas dos tecnologías con el desarrollo del estudio previo.

Una vez realizado el análisis comparativo entre las dos arquitecturas, tanto WCF como Web Services, y con la posterior implementación de las dos tecnologías, se puede afirmar que los objetivos iniciales del proyecto han sido cumplidos totalmente. La realización de un análisis exhaustivo entre las tecnologías con sus diferencias teóricas y sus diferentes características de implementación confirman la correcta ejecución de todos los objetivos de este proyecto.

Como dificultad de los objetivos conseguidos, mencionar la implementación de las dos tecnologías que son muy utilizadas actualmente por ser dos de los referentes de las arquitecturas orientadas a servicio. De la misma forma, las funcionalidades del proyecto han sido implementadas de la misma forma en las dos aplicaciones, lo cual no resta la dificultad de implementar todos los requisitos expuestos en la fase de análisis de este proyecto.

## 10 Conclusiones

En la finalización de este proyecto se pueden extraer diferentes conclusiones, sobre los objetivos del mismo, sobre las tecnologías utilizadas, sobre la propia gestión del proyecto y personales.

Respecto a los objetivos del mismo se ha dedicado todo un apartado, el cual se puede resumir con la finalización y cumplimiento de todos los objetivos propuestos inicialmente.

Las tecnologías utilizadas han sido una parte interesante en el desarrollo de este proyecto, ya que actualmente las arquitecturas orientadas a servicio son muy utilizadas en las implementaciones reales. Todo el proyecto se ha basado en el análisis de las características de estas tecnologías así como la implementación de estas.

Windows Comunication Foundation (WCF) es un conjunto de librerías facilitado por Microsoft en el Framework 3.0, que facilitan la creación de este tipo de arquitecturas orientadas a servicio. Pero no solo ofrecen la posibilidad de la implementación de estas tecnologías sino las características que ofrecen a estas arquitecturas como son la seguridad, escalabilidad y la privacidad de las comunicaciones. La tecnología WCF ofrece la facilidad de implementar una solución en una tecnología como puede ser SOAP y XML, y poder transformar o convertir esta implementación en otro tipo de tecnología cómo puede ser el .NET Remoting o MSMQ. Por lo tanto, encapsula el tipo de solución para facilitar la implementación de las diferentes arquitecturas.

La contrapartida del uso de las Windows Comunication Foundation es la interoperabilidad, ya que en este caso los Web Services ofrecen una total interoperabilidad con los diferentes sistemas ya que es un sistema abierto, donde las especificaciones están abiertas y son implementadas por todos los sistemas. Por otro lado, las Windows Comunication Foundation únicamente ofrecen esta interoperabilidad para

poder ofrecer la disponibilidad de sus servicios en modo SOAP, mientras que el .NET Remoting, MSMQ son especificaciones privadas que únicamente pueden ser implementadas por medio de esta tecnología.

El conjunto de soluciones que ofrecen las Windows Comunication Foundation supone un gran avance y una gran facilidad para la implementación de una misma solución en diferentes arquitecturas, así como diferentes soluciones en una misma arquitectura.

La gestión del proyecto ha sido otra importante tarea realizada con éxito, ya que como se analizo en el inicio de este, existían varios riesgos con una relevancia muy alta dentro del proyecto. Los riesgos han ocurrido con más o menos medida según la planificación inicial, y las acciones para su mitigación han sido llevadas a cabo con éxito con el fin de realizar este proyecto con éxito.

Personalmente este proyecto aporta una consolidación de varios conceptos tratados durante toda la ingeniería, desde conocimientos de diseño de arquitecturas distribuidas, como implementación adecuada en las diferentes capas, desde accesos a base de datos hasta el diseño de las interfaces. También resaltar los conocimientos específicos que me han aportado este proyecto en términos de programación orientada a servicios como el uso de la librería para la implementación de esta tecnología como es Windows Comunication Foundation.

# 11 BIBLIOGRAFÍA

## 11.1. Libros

.NET Application Architecture Guide v2.0

Wrox - Professional WCF 4

Apress – Pro WCF Practical Microsoft SOA Implementation

## 11.2. Enlaces Web

http://msdn.microsoft.com/en-us/library/aa738737.aspx

http://msdn.microsoft.com/en-us/library/aa702755(v=VS.85).aspx

http://msdn.microsoft.com/en-us/library/ms730017(v=VS.85).aspx

http://msdn.microsoft.com/en-us/library/ms734776(v=VS.85).aspx

http://msdn.microsoft.com/en-us/library/bb310550.aspx

http://msdn.microsoft.com/en-us/library/ms730214.aspx

http://msdn.microsoft.com/en-us/library/aa738742.aspx

http://msdn.microsoft.com/en-us/library/ms734776.aspx

http://www.thearchitect.co.uk/presentations/interoperability/WCF-and-Interoperability-Overview.ppt

http://msdn.microsoft.com/es-es/library/ms733128(v=vs.90).aspx

http://msdn.microsoft.com/es-es/library/cc304695.aspx