



FACULTAD DE INFORMÁTICA Y CIENCIAS APLICADAS

TÉCNICO EN INGENIERÍA DE SOFTWARE



ELABORACIÓN DE UN MATERIAL DIDACTICO DE APOYO A LA
ASIGNATURA DE DESARROLLO DE SISTEMAS INFORMATICOS WEB II
PARA EL DESARROLLO DE COMPETENCIAS EN LA CATEDRA DE
DESARROLLO DE SISTEMAS.

TRABAJO DE GRADUACION PRESENTADO POR:

CHÁVEZ CASTRO, KELIN MADAHI

MENDEZ VILLALOBOS, FROYLAN BALMORIS

PARA OPTAR AL GRADO DE:

TECNICO EN INGENIERIA DE SOFTWARE

SEPTIEMBRE, 2012

SAN SALVADOR, EL SALVADOR, CENTROAMERICA

LIC. JOSÉ MAURICIO LOUCÉL

RECTOR

ING. NELSON ZÁRATE SÁNCHEZ

VICERRECTOR GENERAL

LICDA. LISSETTE CRISTALINA CANALES DE RAMÍREZ

DECANO

JURADO EXAMINADOR

ING. TOMAS EDUARDO URBINA

PRESIDENTE

LIC. FRANCISCO ROLANDO TORRES

PRIMER VOCAL

ING. MARLON GIOVANNI MARTÍNEZ PÉREZ

SEGUNDO VOCAL

SEPTIEMBRE, 2012

SAN SALVADOR, EL SALVADOR, CENTROAMERICA

En esta oportunidad quiero agradecer principalmente a Dios por ser él quien ha permitido que llegue a culminar mi carrera con sabiduría y entendimiento, porque sin él nada de esto se hubiera realizado, también quiero agradecer a mis padres Santos Antonio Chávez (QEPD) y María Esperanza Castro por ser un pilar fundamental en mi vida, apoyándome a lo largo de mi carrera, siempre deseándome lo mejor, agradecer a esas personas especiales que han estado conmigo apoyándome tanto económicamente como también con sus consejos y siempre enseñándome a alcanzar mis metas (Rosa Garay, Nancy Berrios, Alexandra Cabas e Irina Flores).

También quiero agradecer a mis herman@s y amig@s por comprenderme en los momentos de tensión a lo largo de la carrera y a esos docentes que siempre me motivaron a seguir y luchar por lograr los objetivos de cada asignatura recibida principalmente al Ing. Edwin Alfredo Callejas que gracias a su enseñanza y conocimientos se ha elaborado el Proceso de Graduación para poder culminar con éxito mi carrera.

Atentamente: Kelin Madahí Chávez Castro

Quiero agradecer principalmente a Dios por prestarme la vida, darme el tiempo, sabiduría y la economía necesaria para poder culminar la primer meta en mis estudios superiores, a mis padres Carlos Méndez y Heriberta Villalobos por el apoyo incondicional y los valiosos consejos que me llenaron de sabiduría en el momento mas preciso, a mi esposa Evelyn Patricia Alfaro por brindarme ayuda y ser paciente en los momentos mas estresantes de mi carrera.

Igualmente quiero agradecer a mi asesor por brindarme los instrumentos y las herramientas necesarias para tener un mejor desempeño en el desarrollo de mi proceso de graduación y poder lograr en mí un estudiante más competente y capaz.

Atentamente: Froylan Balmoris Méndez Villalobos.

INDICE

INTRODUCCION.....	i
CAPÍTULO I: SITUACION ACTUAL.....	3
1.1. SITUACIÓN PROBLEMÁTICA	3
1.2. JUSTIFICACIÓN	4
1.3. OBJETIVOS.	6
1.3.1. OBJETIVO GENERAL:.....	6
1.3.2. OBJETIVOS ESPECIFICOS:.....	6
1.4. ALCANCES	6
1.4.1. GEOGRÀFICA:.....	6
1.4.2. TEMPORAL:.....	7
1.4.3. ORGANIZACIÓN:.....	7
1.4.4. FUNCIONAL:.....	7
1.5. ESTUDIO DE LAS FACTIBILIDADES	8
CAPÍTULO II: DOCUMENTACIÓN TÉCNICA.....	10
2.1. Recopilación de Requisitos.....	10
2.2. Recopilación de requisitos Técnicos.....	12
2.3. Instalación y configuración de herramientas.	34
CAPÍTULO III: DISEÑO DE LA SOLUCIÓN.....	50
3.1. Diseño del Material Didáctico.	50
3.2. Diseño de Ejemplos Prácticos.	56
3.3. Diseño de ejercicios Propuestos.....	73
RECOMENDACIONES.....	84
CONCLUSIONES	85
BIBLIOGRAFIA.....	86
ANEXO 1: CRONOGRAMA DE ACTIVIDADES	88
ANEXO 2: PRESUPUESTO.....	89
ANEXO 3: ENTREVISTA #1 (Dirigida a Estudiantes de Desarrollo de Sistemas) ...	91
ANEXO 4: ENTREVISTA #2 (Dirigida a Estudiantes de Desarrollo de Sistemas)	92
ANEXO 5: ENTREVISTA #3 (Dirigida a Docentes de Desarrollo de Sistemas)	93

INTRODUCCION

Para que todo estudiante de la carrera de Técnico en Ingeniería de Software sea de alto rendimiento y se desempeñe como un profesional de éxito se necesita que las bases, temáticas a implementar en el desarrollo y aprendizaje de las asignaturas, para que le puedan ayudar al estudiante a desarrollar su potencial en ese sentido; por esa razón, se hace necesario la implementación de un material didáctico de apoyo a la asignatura de Desarrollo de Sistemas Informáticos Web II para el desarrollo de competencias en la Cátedra de Desarrollo de Sistemas, el cual ayudará a los estudiantes a desempeñarse con más conocimientos en cuanto a los temas de la asignatura.

En el presente documento se mostrará un bosquejo de lo que se pretende desarrollar de manera factible, ya que en el proceso se elaborará un resumen sobre la situación problemática actual en la asignatura, de tal forma se evaluarán los diferentes problemas a los cuales se les pretende dar solución, luego se justificará el porqué del proyecto, donde se plantearán las diferentes soluciones a cada problemática, se realizarán objetivos en los cuáles se generalizará sobre la meta real del proyecto y objetivos específicos donde se hará una visión más clara de lo que se plantea y de lo que se pretende solucionar, conociendo todo lo anterior se formará el alcance que tendrá el proyecto, definiendo sus límites, luego se realizará el estudio de la factibilidad técnica, económica, operativa y legal con conocimientos sobre los

requerimientos que serán necesarios para el desarrollo de las temáticas, para concluir con el proyecto se dará marcha a la elaboración del cd interactivo que comprende de la elaboración de videos mostrando el desarrollo de las unidades que existen en la asignatura, también dentro de este guías de ejercicios como también ejemplos prácticos que le servirán al estudiante a saber guiarse para poder realizar sus practicas dentro de la asignatura como a lo largo de su carrera, ya para finalizar se realizará un cronograma de actividades que permitirá llevar un orden de todo lo planteado en este documento y el presupuesto de los costos necesarios con los que se ha contado para la elaboración de este material.

CAPÍTULO I: SITUACION ACTUAL

1.1. SITUACIÓN PROBLEMÁTICA

Desarrollo de Sistemas Informáticos Web II es una asignatura impartida a los Técnicos en Ingeniería de Software, dicha asignatura es un pilar en el aprendizaje del estudiante, la cual cuenta con una gama de enseñanzas que hacen el desempeño de las asignaturas en busca de la excelencia, sin embargo, una de las necesidades que influyen en el desarrollo eficiente, es la falta de documentos o manuales de ayuda para los estudiantes. Por muy mínimo que este detalle parezca, es muy importante para el aprendizaje del estudiante y actualmente no se posee ninguna forma de ayuda en la cual éste pueda apoyarse.

La Cátedra de Desarrollo de Sistemas cuenta con asignaturas orientadas al mejor aprendizaje del estudiante las cuales son: Base de Datos I y II, Programación I, II y III, Ingeniería de Software, Arquitectura de Software, Desarrollo de sistemas Informáticos Web I y II; asignaturas con las que el estudiante alcanza amplios conocimientos, por lo tanto la asignatura Desarrollo de Sistemas Informáticos Web II (DSIW2-T) tiene como propósito la unión de todas estas distintas asignaturas, la cual forma un conocimiento muy amplio y un desarrollo extenso, dicha asignatura se forma a partir de los conocimientos adquiridos en las otras asignaturas, dadas en la cátedra, donde se les da seguimiento y refuerzo a lo anteriormente aprendido, actualmente en las

bibliotecas físicas y bibliotecas virtuales hay muchos libros y documentos que sirven de ayuda a las asignaturas de Base de Datos, Programación, Estándares de la Programación, pero estos resuelven sus diferentes problemáticas, temas u objetivos y se desarrollan temáticas muy propias de la asignatura y raramente unen las distintas tecnologías y las enseñanzas impartidas en las asignaturas, tomando en cuenta que no existe ningún documento o libro que tome todo lo necesario del aprendizaje de las asignaturas y lo una haciendo uso global de todas las enseñanzas apuntando al desarrollo de sistemas, esta es la razón más clara por la cual se elaborará un material didáctico, el cual saque la esencia de la enseñanza de las distintas asignaturas, enfocadas en la unión de las tecnologías y desarrollo para resolver las problemáticas de la asignatura, el documento hará la unión de todas las enseñanzas y hará más fácil el aprendizaje del estudiante, las enseñanzas del docente y por ende el paso por esta asignatura será más gratificante para el estudiante.

1.2. JUSTIFICACIÓN

Es indiscutible y sumamente necesario la utilización de distintas herramientas, documentos y manuales de ayuda, para la elaboración de cualquier proyecto en clases y para elaborar tareas ex aulas, por tal razón, el presente trabajo parte de la necesidad de tener esas referencias y manuales didácticos para aumentar la probabilidad de superación y triunfo en el aprendizaje del estudiante. En este material lo que se busca es lograr un amplio conocimiento

usando como base las enseñanzas del docente y como apoyo, las distintas formas de ayuda que se presentan a la materia y al estudiante.

Actualmente ya no es muy común que los estudiantes asistan a la biblioteca ya sea físicas o virtuales en búsqueda de información basadas en referencias bibliográficas, normalmente los estudiantes hacen uso de la fuente de búsqueda más grande del mundo que es el internet por medio de una PC para obtener la información actualizada y concisa en cuestiones de minutos, al utilizar Internet como herramienta de comunicación se amplía el radio de cobertura de los desarrollos de investigación, permitiendo una mayor accesibilidad a la información; sin embargo, esto también incluye un costo económico elevado para su acceso y suele ser algo tedioso y molesto cuando no se sabe buscar la información necesaria o la información no es la adecuada, esto forma parte de uno de los contra en el aprendizaje del estudiante.

El documento a presentar nace a partir de la necesidad de un material didáctico que sirva de referencia al estudiante de la asignatura de Desarrollo de Sistemas Informáticos Web II, donde se pondrá en funcionamiento una cantidad de manuales que servirá para acompañar las distintas enseñanzas que el docente imparta y así mismo en el desarrollo de la asignatura.

1.3. OBJETIVOS.

1.3.1. OBJETIVO GENERAL:

- Elaborar un material didáctico de apoyo de competencias para los estudiantes de la asignatura de Desarrollo de Sistemas Informáticos Web II de la Cátedra de Desarrollo de Sistemas.

1.3.2. OBJETIVOS ESPECIFICOS:

- Realizar una investigación bibliográfica y de campo sobre los contenidos que se desarrollan en la asignatura para elaborar el material didáctico.
- Configurar las plataformas de desarrollo para que el material de apoyo sea ejecutado por los estudiantes con modelos y manuales.
- Realizar un cd interactivo de las diferentes temáticas a desarrollar en la asignatura.

1.4. ALCANCES

1.4.1. GEOGRÀFICA:

El proyecto se desarrollará e implementará en la tercera planta de edificio Gabriela Mistral de la Universidad Tecnológica de El Salvador, con uso exclusivo de la Facultad de Informática y Ciencias Aplicadas.

1.4.2. TEMPORAL:

El material se desarrollará en el periodo comprendido de cuatro meses y este será de gran importancia para los estudiantes que realizarán sus actividades implementando el material interactivo. La proyección se basa en las temáticas que se esperan alcanzar para la asignatura y que los estudiantes tengan más facilidad en su aprendizaje y desempeño realizando y aplicando las funciones que se le enseñen.

1.4.3. ORGANIZACIÓN:

Este material se elaborará en función del equipo de trabajo que proporcionará la información al estudiante contando con el asesor del material didáctico que corresponde al Ing. Marlon Giovanni Martínez Pérez, como también el metodológico que será proporcionado por el Ing. Edwin Alberto Callejas y por los estudiantes Kelin Madahi Chávez Castro y Froylan Balmoris Méndez Villalobos.

1.4.4. FUNCIONAL:

Se espera que como estudiantes de la Carrera de Técnico en Ingeniería de Software los estudiantes logren tener un mejor aprendizaje y mejores opciones para desarrollarse y obtener mejores resultados en su desempeño.

- ✓ Desarrollo de un cd interactivo.

- ✓ Material didáctico de la asignatura, basado en el DI (Diseño Instruccional)
- ✓ Ejemplos prácticos de cada contenido.
- ✓ Propuesta de ejercicios.

1.5. ESTUDIO DE LAS FACTIBILIDADES

➤ TÉCNICA:

Los materiales necesarios para implementar este material son acordes para su desarrollo, ya que la Universidad cuenta con laboratorios en los cuales los estudiantes pueden trabajar en la elaboración del material didáctico así como en el cd-interactivo que se proporcionará al estudiante, cada laboratorio cuenta con computadoras de diferentes características, entre ellas están: **(ver figura 1, Características de la Pc para el desarrollo del material)**.

Atributo	Descripción
Disco Duro (HD)	120 Gb.
Memory Ram	4 Gb.
Microprocesador	AMD 3 GHz Core Duo.
Tarjeta Red	10/100mb.
Sistema Operativo	Windows 7 Ultimate 32- bit.
IDE	Visual Studio 2010 Profesional 86x.
Sistema Gestor de Base de	SQL Server 2008 Developer 86x.

Datos (SG/BD)	
Tarjeta de video	1024 Mb.
Red	Topología de estrella de 4 Mb de velocidad por segundo.

Figura 1, Características de la Pc para el desarrollo del material.

➤ **ECONÒMICA:**

Gracias a este material de apoyo muchos estudiantes estarán más preparados con su aprendizaje. La elaboración del mismo no tendrá mayor inversión económica, y se realizará como parte del trabajo de graduación.

➤ **OPERATIVA:**

Este material de apoyo interactivo es de gran importancia para los estudiantes ya que es factible su uso, puesto que ellos han llevado un proceso de aprendizaje de diferentes asignaturas con diversas temáticas; por lo cual tienen los conocimientos y pueden usarlo más fácilmente y hacer mas factiblemente su desarrollo en la asignatura.

➤ **LEGAL:**

La Universidad Tecnológica cuenta con convenios para el desarrollo de competencias por tal razón no hay por qué restringir la herramienta a desarrollarse para el uso por parte de los estudiantes.

CAPÍTULO II: DOCUMENTACIÓN TÉCNICA

2.1. Recopilación de Requisitos.

Para comenzar con la documentación necesaria para el desarrollo del material didáctico es necesario realizar recopilación de requisitos de hardware, software e infraestructura de red. Se comenzará conociendo los requisitos de instalación y ejecución del software de diversas plataformas, para un excelente funcionamiento de las aplicaciones a desarrollar es necesario que se cuenten con instrucciones, entre ellas están: Instalación de Microsoft Visual Studio 2010 Professional x86, SQL Server 2008 Developer 86 e Instalación y Configurar del IIS (Internet Information Services).

2.1.1. Requisitos de Instalación de Microsoft Visual Studio 2010 Professional.

Las herramientas de implementación de Microsoft Visual Studio 2010 incluyen la capacidad de detectar automáticamente la existencia de componentes durante la instalación e instalan un conjunto de requisitos previos predeterminado.

Microsoft Visual Studio 2010 se puede instalar mediante los discos de instalación (por ejemplo, DVD) o descargándolo del sitio web del producto, entre sus requisitos están:

Mediante los discos de instalación:

Para instalar Microsoft Visual Studio 2010, ejecute primer Setup.exe, que se encuentra en el directorio raíz de los discos de instalación de Microsoft Visual Studio 2010.

Mediante descarga del sitio web del producto:

Esta instrucción se puede ejecutar ingresando al sitio web de descargas de Microsoft Visual Studio 2010, seleccionando la edición que se desee descargar.

2.1.2. Requisitos de Instalación de SQL Server 2008 Developer.

SQL Según la herramienta que se vaya a instalar se deben tener en cuenta los requerimientos mínimos de cada una de estas para evitar problemas al momento de la instalación y ejecución de la misma entre ellas están, **(ver tabla 1, Requisitos mínimos del hardware para instalación para SQL Server 2008 Developer)**.

Componente	Requerimiento
Procesador	Mínimo: 1.4 GHz (x64).
Memoria	Mínimo: 512 MB RAM - Máximo: 8 GB.
Espacio en Disco	Mínimo: 32 GB o superior.

Pantalla	Súper VGA (800 × 600) o monitor con resolución superior.
Otros	Unidad DVD, Teclado, dispositivo apuntador, conexión a internet.

Tabla 1: Requisitos mínimos del hardware para instalación para SQL Server 2008 Developer.

2.1.3. Requisitos de Instalación del Internet Information Services (IIS):

La instalación de IIS 7.0 prevé algunos requisitos tales como: la presencia del Service Pack 3 o superior y de Explorer 4 o superior.

IIS7.0 requiere la instalación de Windows NT 4.0 correctamente configurado e instalado en una plataforma hardware potencialmente sobre-dimensionada.

- Intel Pentium 100MHz, de un mínimo de 32 MB a 64 MB RAM.
- 60 MB de espacio en disco duro para la instalación. De un mínimo de 250MB a 2GB de espacio libre en el disco duro para el Caching.

2.2. Recopilación de requisitos Técnicos.

Tiempo atrás no se contaba con la informática tan avanzada como hoy, en esa época las personas tenían que ingeniárselas para poder desarrollar algún sistema o ejecutar diferentes procesos que eran de alto costo y en algún momento de su vida muchos proyectos no se llegaban a concluir por no contar

con los recursos necesarios, hoy en día la tecnología es una de las virtudes con las que se cuenta en el mundo y a la cual todos y todas estamos en la disposición de adquirirla, el área de informática ha venido desarrollándose de tal manera que existe mucha información que es de alguna manera algo tediosa por las ciencias y tecnologías que se aplican a cada temática a desarrollarse, la asignatura de Desarrollo de Sistemas Informáticos Web II es una asignatura en la que muchos estudiantes tienen un alto en su carrera por abarcar muchos temas en los cuales no se cuenta con una mejor resolución de diversas temáticas, es una asignatura que juega un papel muy importante en el desarrollo de los estudiantes que están cursando esta asignatura por tal razón es indispensable conocer paso a paso su desarrollo y temas a desarrollar de la misma, es por eso que se debe conocer su metodología, diseño e instrucciones a seguir para un mejor desempeño de las diferentes plataformas que forman la asignatura, preguntarse ¿cómo será su desarrollo?, ¿cómo será esa temática? y ¿qué es lo que se necesita para llevar a cabo el material didáctico?; lo que le ayudará a los estudiantes a desarrollarse como programadores y conocer así todas las disposiciones con las que se cuenta en la asignatura.

Para llevar a cabo la resolución del material didáctico se debe tener en cuenta que es lo que se necesita y sus requisitos por esa razón se habla de los instrumentos a implementar para elaborarlo, como se conoce en la rama de la informática y en la asignatura el desarrollo de esta debe contar con ciertos programas que determinen el desarrollo de sistemas para que el estudiante

lleve en práctica su teoría, entre ellos están los requisitos que asp.net cuenta y conocer previamente ¿qué es asp.net? ¿Para qué sirve? ¿En qué consiste? porque se ha comprobado que muchos estudiantes llegan a cursar la asignatura de Desarrollo de Sistemas Informáticos Web II y algunos aun no saben en que consiste este framework por tal razón se especifica y se conceptualiza.

¿Qué es asp.net? Es un framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework.

Cualquier persona que está familiarizada con el desarrollo de aplicaciones web sabrá que el desarrollo web no es una tarea simple. Ya que mientras que un modelo de programación para aplicaciones de uso común está muy bien establecido y soportado por un gran número de lenguajes, herramientas de desarrollo, la programación web es una mezcla de varios lenguajes de etiquetas, un gran uso de lenguajes de script y plataformas de servidor. Para el programador de nivel intermedio, el conocimiento y las habilidades que se necesitan para desarrollar aplicaciones web requieren de un mayor conocimiento tanto de lenguajes de programación, etiquetado y formato, como de diversas tecnologías de software relativas al desarrollo distribuido y

concurrente, de las que son necesarias en el desarrollo tradicional de aplicaciones.

La asignatura de Desarrollo de Sistemas Informáticos Web II cuenta con una gama de unidades a las cuales se le derivan sus temas para desarrollarse que son de mucho interés en el proceso del desarrollo de la asignatura, para ver dichas unidades; **(Ver Tabla 2, Unidades de la Asignatura de Desarrollo de Sistemas Informáticos Web II).**

UNIDAD	NOMBRE	DESCRIPCION
Unidad I	Desarrollo de Sistemas Informáticos Transaccionales	Desarrollar sistemas de información transaccionales de apoyo a procesos operativos del negocio.
Unidad II	Desarrollo de Sistemas Informáticos con Membership	Incorporar en el desarrollo de S.I transaccionales, la seguridad integrada de .Net a través de Membership.
Unidad III	Desarrollo de Sistemas Informáticos con tecnología Linq	Desarrollar sistemas de información transaccionales utilizando la tecnología LINQ para la construcción ágil y robusta de sistemas.
Unidad IV	Desarrollo de Sistemas Informáticos utilizando Clases y Web Services	Utilizar clases y web Services para lograr la interoperabilidad entre sistemas.

Unidad V	Desarrollo de Sistemas Informáticos utilizando Ajax	Utilizar Ajax para agregarle contenido enriquecido y usabilidad a las aplicaciones web.
----------	---	---

Tabla 2: Unidades de la Asignatura de Desarrollo de Sistemas Informáticos Web II.

Cada unidad está estructurada por sus temas a desarrollarse, que le facilitan tener el conocimiento necesario sobre la asignatura y así poder permitir una mejor satisfacción de entendimiento aplicando una o diferentes metodologías de desarrollo para los estudiantes, en conclusión se comenzará a explicar cada tema de las diferentes unidades que se desarrollan.

2.2.1. Unidad I: Sistemas de Información Transaccionales.

2.2.1.1. Sistemas Transaccionales: (Transactional System, sistema de procesamiento de transacciones).

Es un tipo de sistema de información diseñado para recolectar, almacenar, modificar y recuperar todo tipo de información que es generada por las transacciones en una organización. Una transacción es un evento o proceso que genera o modifica la información que se encuentran eventualmente almacenados en un sistema de información.

Función concreta de un sistema transaccional

Un sistema transaccional debe controlar las transacciones para mantener la seguridad y consistencia de los datos involucrados. Por ejemplo, un cliente

transfiere dinero de una cuenta a otra cuenta dentro de un mismo banco; la cantidad de dinero que se descuenta de la cuenta emisora debe ser igual a la que se suma en la cuenta receptora. De no ser así, la acción (transacción) no se realiza.

Un sistema transaccional debe ser capaz de enmendar cualquier error ocurrido durante una transacción, pudiendo deshacer las operaciones realizadas, manteniendo los datos tal cual estaban antes del error.

También debe ser capaz de controlar y administrar múltiples transacciones, determinando prioridades entre éstas. Por ejemplo, un cliente está haciendo la reserva de un asiento en un vuelo, dicho asiento debe ser bloqueado temporalmente hasta que se concrete la transacción, porque otro cliente podría estar queriendo reservar el mismo asiento en el mismo momento.

Tipos de decisiones: los especialistas en el análisis del sistema debe verificar cada paso o decisión a tomar que será de gran importancia para el desarrollo de estos, debe planear estrategias que sirvan de control para los procesos adquiridos a lo largo del ciclo, un ejemplo claro es: **(Ver figura 3, Tipos de decisiones)**.



Figura 1: Tipos de decisiones.

Propiedades de los Sistemas Transacciones:

Los sistemas transaccionales cuentan con diversas propiedades las cuales son de gran importancia para desarrollar, entre ellas están. (Ver figura 2, Propiedades de los Sistemas Transaccionales).

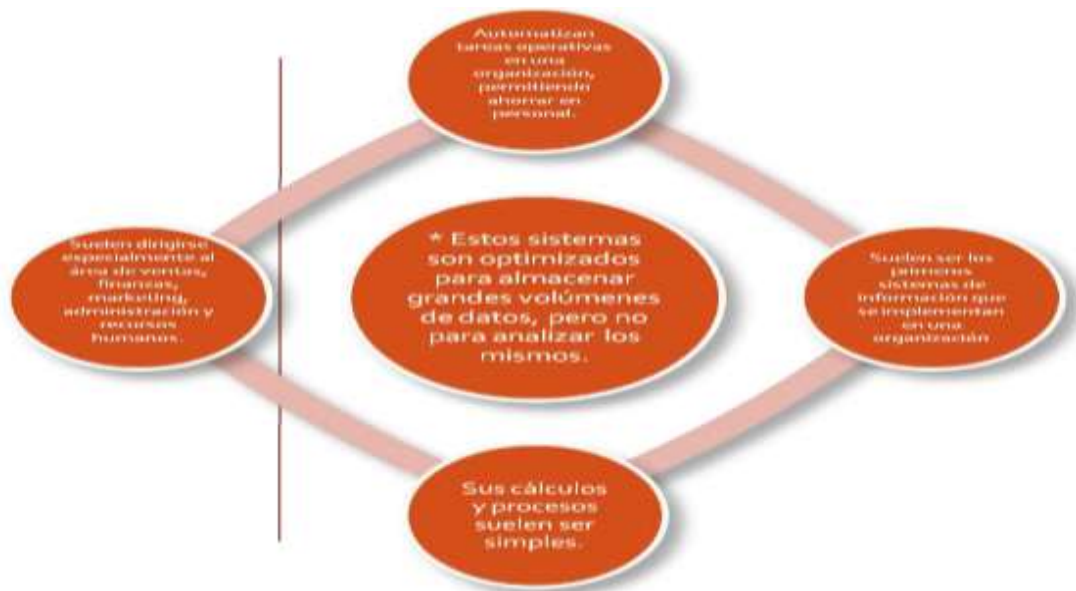


Figura 2: Propiedades de los Sistemas Transaccionales.

Características:

Para que un sistema informático pueda ser considerado como un sistema transaccional, debe superar el test ACID, estos deben de controlar las transacciones para mantener la seguridad y consistencia de los datos involucrados.

- Rapidez: deben ser capaces de responder rápidamente, en general la respuesta no debe ser mayor a un par de segundos.
- Fiabilidad: deben ser altamente fiables, de lo contrario podría afectar a clientes, al negocio, a la reputación de la organización, etc. En caso de fallas, debe tener mecanismos de recuperación y de respaldo de datos.
- Inflexibilidad: no pueden aceptar información distinta a la establecida.

Los Sistemas Transaccionales cuentan con tres niveles que son necesarios en su desarrollo. Son los procesos que le dan soporte a los sistemas. (**Ver figura 3, Niveles de los Sistemas Transaccionales**).



Figura 3: Niveles de los Sistemas Transaccionales.

- Sistemas Gerenciales.
- Sistemas de Apoyo a la toma de decisiones.
- Generación de Reportes ad-hoc.
- Existe gran volumen de procesamiento de datos.
- Existe gran cantidad de consulta de datos.

2.2.2. Unidad II: Desarrollo de Sistemas Informáticos con Membership.

Membership: Es la suscripción de ASP.NET que le proporciona una forma integrada para validar y almacenar las credenciales del usuario. La suscripción (Membership) de ASP.NET por lo tanto, le ayuda a gestionar la autenticación de usuario en los sitios Web. Puede utilizar la suscripción de ASP.NET con ASP.NET de autenticación de formularios mediante el uso de la conexión controles ASP.NET para crear un sistema completo para la autenticación de usuarios. La suscripción de ASP.NET soporta instalaciones para: Creación de nuevos usuarios y contraseñas.

El almacenamiento de la información de suscripción (nombres de usuario, contraseñas y datos de apoyo) en Microsoft SQL Server, Active Directory o un almacén de datos alternativo. La autenticación de usuarios que visitan su sitio. Puede autenticar a los usuarios mediante programación, o puede utilizarlos controles ASP.NET de inicio de sesión para crear un sistema de autenticación completo que requiera poco o ningún código.

La gestión de contraseñas, que incluyen la creación, modificación y reposición esto dependiendo de las opciones de membresía que el usuario elija, el sistema de suscripciones también puede proporcionar un sistema automatizado de restablecimiento de contraseña, el sistema que tiene una pregunta proporcionada por el usuario y la respuesta.

La exposición de una identificación única para los usuarios autenticados que se pueden utilizar en sus propias aplicaciones y que también se integra con la personalización de ASP.NET y la función de gestión (autorización) de sistemas. Especificación de un proveedor de suscripciones personalizado, que le permite sustituir su propio código para administrar la pertenencia y mantenerlos datos de afiliación en un almacén de datos personalizado.

Roles y propiedades de Membership:

Las afiliaciones de una característica autónoma en ASP.NET para la autenticación, se puede integrar con la administración de funciones de ASP.NET para proporcionar servicios de autorización para su sitio. La membresía también puede ser integrada con las propiedades del perfil de los usuarios para proporcionar una aplicación específica de personalización que se puede adaptara los usuarios individuales.

Uso y Configuración de Membership:

Para utilizar Membership, primero debe configurar en el sitio.

Los siguientes pasos son básicos que permiten seguir con la configuración de Membership.

1. Especifique las opciones de membresía como parte de la configuración del sitio Web. De forma predeterminada los miembros están habilitados. También puede especificar qué proveedor de suscripciones desea utilizar. El proveedor de tiendas de membresía por defecto de información en una base de datos de Microsoft SQL Server. Sin embargo, puede utilizar otros proveedores para almacenarla información de suscripción, o puede especificar un proveedor personalizado.
2. Configurar la aplicación para utilizar la autenticación de formularios (a diferencia de Windows). También puede especificar que algunas páginas o carpetas de la aplicación están protegidos y sólo son accesibles a los usuarios autenticados.
3. Definir las cuentas de usuario para la adhesión. Esto se puede hacer en una variedad de maneras. Usted puede utilizar el Sitio Web Herramienta de administración, lo que proporciona una interfaz de tipo asistente para la creación de nuevos usuarios. Alternativamente, puede utilizar una página Web ASP.NET donde se recogen un nombre de usuario y contraseña (y, opcionalmente, una dirección de correo electrónico), y luego usar el método de pertenencia

CreateUser para crear un nuevo usuario en el sistema de suscripciones.

Puede utilizar la suscripción para autenticar a los usuarios en su aplicación. Por lo general, va a proporcionar una página donde los usuarios pueden iniciar sesión en una página de registro donde los usuarios nuevos pueden registrarse, y una página de cambio de contraseña, donde los usuarios pueden cambiar su contraseña. Los controles ASP.NET de inicio de sesión (Entrar, LoginView, LoginStatus, LoginName y PasswordRecovery) encapsulan virtualmente toda la lógica necesaria para solicitar a los usuarios las credenciales y validarlas mismas en el sistema de suscripciones.

La plantilla predeterminada por defecto en Microsoft Visual Studio 2010 para crear un proyecto ASP.NET incluye páginas de gran parte de esta funcionalidad básica de inicio de sesión. Alternativamente, puede utilizar los mismos controles ASP.NET que se encuentran en la plantilla para crear páginas de inicio de sesión personalizado y de membresía.

Si ha configurado la aplicación para utilizarla autenticación de formularios, ASP.NET mostrará automáticamente la página de entrada si un usuario solicita una página protegida no autenticados.

2.2.3. Unidad III: Desarrollo de Sistemas Informáticos con Linq.

2.2.3.1. ¿Qué es Linq? Linq: Language Integrated Query.

Como su nombre lo indica, integra conceptos de consultas directamente en los lenguajes de programación, permitiendo que el código de acceso a datos sea verificado por el compilador y las herramientas de desarrollador, como Intellisense para hacer que los desarrolladores escriban queries.

Como grandes ventajas de Linq están:

1. La unificación del acceso a los datos, ya que es posible cambiar de fuentes de datos sin apenas cambios en LinQ.
2. Confección de consultas más sencillas.
3. Y sobre todo la optimización en los tiempos de codificación y resolución de errores, ya que ahora se puede hacer debug en tiempo de ejecución, al estar completamente integrado con .net framework.

LinQ usa una alternativa que toma la escritura y el modelo de la base de datos relacional y las convierte en un modelo de objetos a través de LINQ to SQL, es decir que LinQ toma las consultas SQL y las convierte en un modelo de objetos y de la misma manera sucede al revés al recibir los resultados de la base de datos LINQ to SQL los vuelve a convertir en objetos, este hecho se realiza para poder trabajarlos en su propio lenguaje de programación.

En cuanto a los procedimientos almacenados estos también se transforman en un modelo de objetos que también pueden ser usados (clases) con los triggers.

Los procedimientos Almacenados en Transact SQL, pueden devolver valores (numérico entero) o conjuntos de resultados.

Para crear un procedimiento almacenado debemos emplear la sentencia “create procedure”.

```
CREATE PROCEDURE <nombre_procedure> [@param1 <tipo>,...]
```

AS

Sentencias del procedure

Para modificar un procedimiento almacenado se debe emplear la sentencia “alter procedure”.

```
ALTER PROCEDURE <nombre_procedure> [@param1 <tipo>,...]
```

AS

Sentencias del procedure

2.2.4. Unidad IV: Desarrollo de Sistemas Informáticos utilizando Clases y Web Services.

2.2.4.1. Web Services.

Un servicio web (en inglés, Web Service) es una pieza de software que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en

lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares.

Los servicios web son la revolución informática de la nueva generación de aplicaciones que trabajan colaborativamente en las cuales el software está distribuido en diferentes servidores.

¿Cómo funcionan en .NET?

Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento. Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

La principal razón para usar servicios Web es que se pueden utilizar con HTTP sobre TCP (Transmission Control Protocol) en el puerto 80. Dado que las organizaciones protegen sus redes mediante firewalls que filtran y bloquean gran parte del tráfico de Internet, cierran casi todos los puertos TCP salvo el 80,

que es, precisamente, el que usan los navegadores. Los servicios Web utilizan este puerto, por la simple razón de que no resultan bloqueados. Es importante señalar que los servicios web se pueden utilizar sobre cualquier protocolo, sin embargo, TCP es el más común.

Microsoft utiliza para conseguir este propósito con su tecnología .NET que se emplea como protocolo de comunicación, una aplicación XML, llamada SOAP.

¿Qué es SOAP? Son las siglas de Simple Object Access Protocol. Este protocolo deriva de un protocolo creado por David Winer, XML-RPC en 1998. En su sitio web, Userland, <http://www.userland.com> se puede encontrar multitud de documentación acerca de este primer protocolo de comunicación bajo http mediante XML. Con este protocolo se podían realizar RPC o remote procedure calls, es decir, podíamos bien en cliente o servidor realizar peticiones mediante http a un servidor web. Los mensajes debían tener un formato determinado empleando XML para encapsular los parámetros de la petición. Con el paso del tiempo el proyecto iniciado por David Winer interesó a importantes multinacionales entre las que se encuentran IBM y Microsoft y de este interés por XML-RPC se desarrollo SOAP.

SOAP es un protocolo más completo que XML-RPC pero cabe decir que más complejo. Otra razón es que, antes de que existiera SOAP, no había buenas interfaces para acceder a las funcionalidades de otros ordenadores en red. Las que había eran ad hoc y poco conocidas, tales como EDI (Electronic Data

Interchange), RPC (Remote Procedure Call), u otras APIs.

Una tercera razón por la que los servicios Web son muy prácticos es que pueden aportar gran independencia entre la aplicación que usa el servicio Web y el propio servicio. De esta forma, los cambios a lo largo del tiempo en uno no deben afectar al otro. Esta flexibilidad será cada vez más importante, dado que la tendencia a construir grandes aplicaciones a partir de componentes distribuidos más pequeños es cada día más utilizada.

Características de Web Service

- ✓ Interoperabilidad:

Permite a las aplicaciones desarrolladas en diferentes lenguajes y corriendo en sistemas heterogéneos comunicarse.

- ✓ Integración Dinámica:

Permite aplicaciones a localizar dinámicamente e integrarse con alguna otra para proveer soluciones empresariales.

- ✓ Estándares de las Industrias:

Permite a las aplicaciones comunicarse con otras usando estándares conocidos como XML, SOAP, WSDL and UDDI.

- ✓ Seguridad:

Permite a las aplicaciones comunicarse bajo un ambiente seguro usando XML signature y XML encryption. La XML

signature y encryption son mecanismos de seguridad para mantener la integridad de los datos que es transferida por internet.

- ✓ Un Web Service puede ser registrado para poder dejarlo a disposición de otros usuarios y para que los mismos puedan localizarlo. Un mecanismo para registrar estos servicios es por medio de UDDI, sigla que corresponde a **Universal Description, Discovery and Integration**, un “repositorio de Web Services”. Para registrar un servicio se tendría que tener en cuenta que se debe suministrar la información de la empresa, en qué categorías se ubicaría el servicio y la interfaz a utilizar para consumir el servicio.
- El mecanismo utilizado por un Web Service para especificar de qué forma hay que proporcionarle los datos, de manera tal que cualquiera pueda interaccionar con el mismo, es por medio de lenguaje **XML**. Esta información se almacena en un archivo llamado WSDL (**Web Services Description Language**), el cual contiene un documento XML junto con la descripción de ciertos mensajes SOAP y cómo deben intercambiarse, así como también dónde está el recurso del servicio y con qué protocolo debe dialogar quien lo consume.
- El protocolo de comunicación utilizado es el **SOAP** generalmente, el cual es relativamente sencillo de utilizar.
- Los Web Services utilizan protocolos comúnmente conocidos y difundidos tales como el formato XML, TCP/IP como protocolo de transporte y HTTP

como protocolo de transferencia de hipertexto, este descrito puede verse en: **(ver figura 4: Protocolos de los Web Services)**



Figura 4: Protocolos de Web Services.

2.2.5. Unidad V: Desarrollo de Sistemas Informáticos utilizando Ajax.

2.2.5.1. Ajax.

Ajax, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dados que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

Ajax es una combinación de cuatro tecnologías ya existentes:

1. XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
2. Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.
3. El objeto XMLHttpRequest para intercambiar datos de forma asíncrona con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar

del XMLHttpRequest para realizar dichos intercambios.

4. XML es el formato usado generalmente para la transferencia de datos solicitados al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBML.

Como el DHTML, LAMP o SPA, Ajax no constituye una tecnología en sí, sino que es un término que engloba a un grupo de éstas que trabajan conjuntamente.

Ventajas:

La experiencia de usuario en la navegación es mucho más rica. Ya no se refresca la página constantemente al interactuar con ella.

- El tiempo de espera para una petición se reduce. El usuario al hacer un pedido (request) al servidor, no se envía toda la página.
- Por la misma razón anterior el tráfico al servidor se reduce.

Desventajas:

- Falta de integración con el botón retroceder del navegador. Se debe tener en cuenta esto al intentar guardar funcionalidad con este botón.

- Falta de soporte para todos los navegadores. Aunque esto se va reduciendo, el problema se presenta por la falta de soporte para JavaScript y XMLHttpRequest.
- Problemas si el usuario ha deshabilitado el uso de JavaScript en su navegador. Hay que tener esto en cuenta cuando desarrollamos nuestro sitio web, para enfrentarnos a esta situación.
- No poder recomendar links específicos. Si hace que toda la web sea interactiva, no podremos recomendar algún link, ya que el contenido fue generado dinámicamente. Se debería encontrar un equilibrio dependiendo de las necesidades de su escenario.
- Demasiado código Ajax hace lento el navegador. A más Ajax, más uso de código JavaScript del lado del browser, por consiguiente mayor trabajo del browser. El rendimiento de la computadora puede verse afectado, por eso no hay que abusar del AJAX.

AJAX en Microsoft Visual Studio y .NET:

AJAX, como mejora para las aplicaciones basadas en web, también es retomada por Visual Studio de Microsoft desde la versión 2005 con el .NET Framework 2.0. Al principio se diseñó una librería strong-typed de .NET para que se agregara como referencia en las aplicaciones web en cualquiera de los lenguajes soportados por este marco de trabajo. Esta DLL se integraba en las aplicaciones y se incluía en el directorio Bin de la aplicación y se distribuía junto con la aplicación final. Después, en las versiones de .NET Framework

superiores, se encontraban agregados algunos controles basados en AJAX que permitían la interactividad, y la librería creada anteriormente ahora se utiliza para funcionalidades específicas como animaciones personalizadas, modal pop ups, paneles animados, etc.

2.3. Instalación y configuración de herramientas.

2.3.1. Instalación de Internet Information Services (IIS).

Objetivo:

Mostrar la instalación de IIS con sus componentes de desarrollo web.

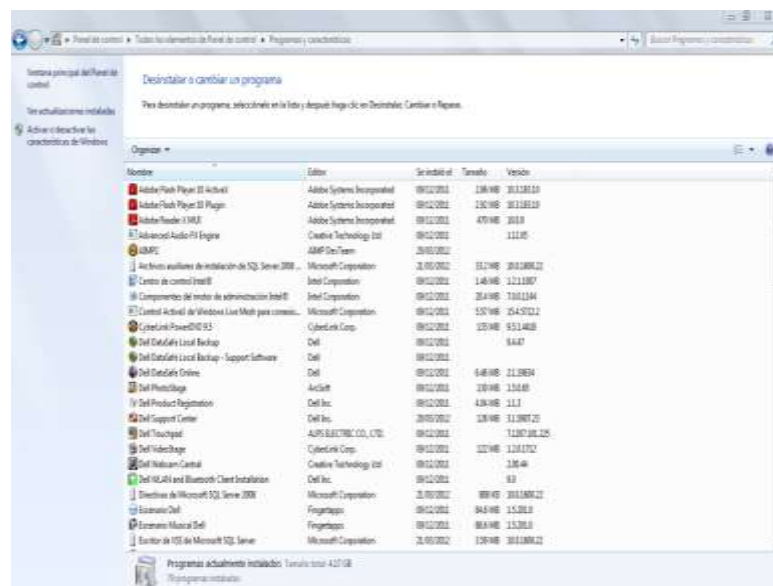


Figura 2.3.1.1. Agregar componentes de IIS.

La figura 2.3.1.1. Muestra la interfaz de agregar nuevos componentes de Microsoft para visualizar el IIS, por lo cual, se selecciona la opción de agregar componentes



Figura 2.3.1.2. Características de IIS.

La figura 2.3.1.2. Muestra la pantalla que permite seleccionar los componentes que sean necesarios para la instalación y configuración.

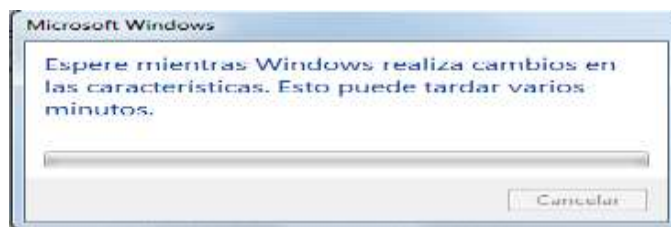


Figura 2.3.1.3. Proceso de instalación.

La figura 2.3.1.3. Muestra el proceso de configuración que se está ejecutando con sus características para la instalación, después de este proceso el sistema pedirá que se realice el reinicio para configurar su entorno.

2.3.2. Instalación de Microsoft Visual Studio 2010.

Objetivo:

Aprender a instalar el software de desarrollo de Microsoft Visual Studio 2010 para la elaboración de proyectos de sistemas de información.



Figura 2.3.2.1. Pantalla de inicio para la instalación.

La figura 2.3.2.1. Es la primera interfaz que existen dos opciones en las cuales se seleccionara la opción de Instalar Microsoft Visual Studio 2010.



Figura 2.3.2.2. Asistente de instalación.

La figura 2.3.2.2. Muestra el asistente que permitirá el proceso de instalación del programa por lo cual se selecciona la opción siguiente.

La figura 2.3.2.4. Es en la que el usuario debe seleccionar las características que se desean instalar y especificar la ruta de acceso en la cual se instalara el programa, posteriormente se selecciona la opción instalar para que dé inicio el proceso de instalación.

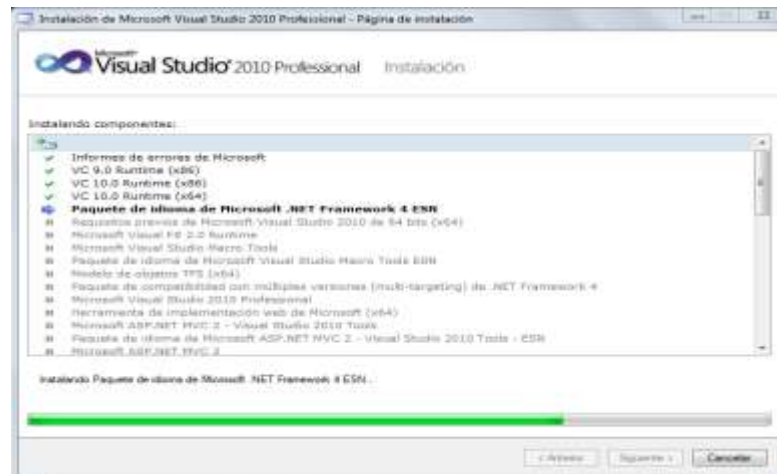


Figura 2.3.2.5. Proceso de ejecución.

La figura 2.3.2.5. Es la interfaz que muestra el proceso final de la instalación en la que se tiene un tiempo determinado para que se ejecute ya que instala todas las características que fueron seleccionadas para la instalación, terminando este tiempo se selecciona Finalizar.

2.3.3. Instalación de SQL Server 2008.

Objetivo:

Ayudar al estudiante a instalar el entorno de desarrollo de SQL Server 2008 para limitar los errores posibles.

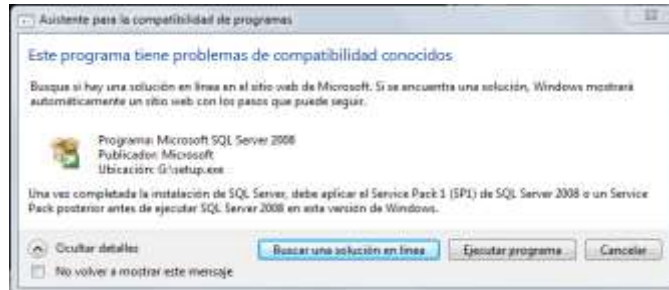


Figura 2.3.4.1. Asistente para la compatibilidad.

La figura 2.3.4.1. Muestra la interfaz de compatibilidad con el sistema en la cual se selecciona ejecutar programa.



Figura 2.3.4.2. Centro de instalación de Sql Server 2008.

La figura 2.3.4.2. Muestra las opciones de instalación, mantenimiento y configuración de las herramientas de Sql Server 2008.



Figura 2.3.4.3. Reglas auxiliares del programa de instalación.

La figura 2.3.4.3. Muestra el cumplimiento de los requisitos para la instalación inicial de Sql Server 2008.

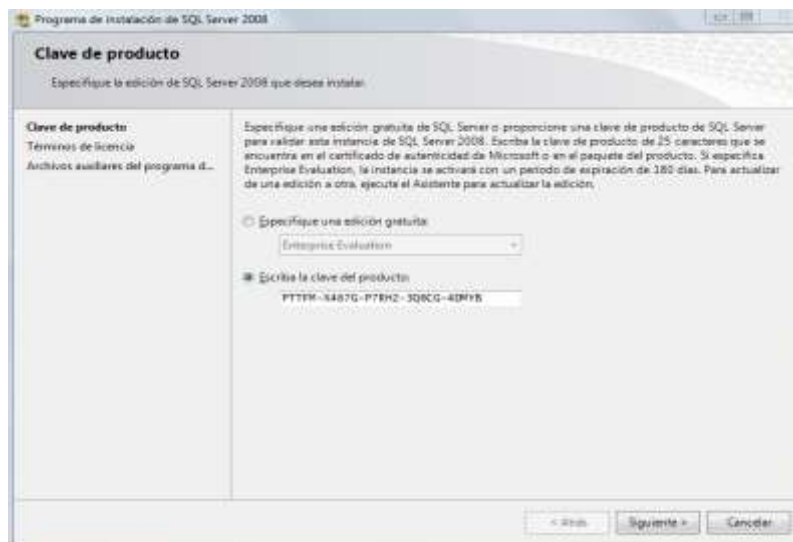


Figura 2.3.4.4. Clave del producto.

La figura 2.3.4.4. Requiere la especificación de la edición del producto a instalar.



Figura 2.3.4.5. Términos de la licencia.

La figura 2.3.4.5. Muestra el términos de licencia de instalación de Sql Server 2008.

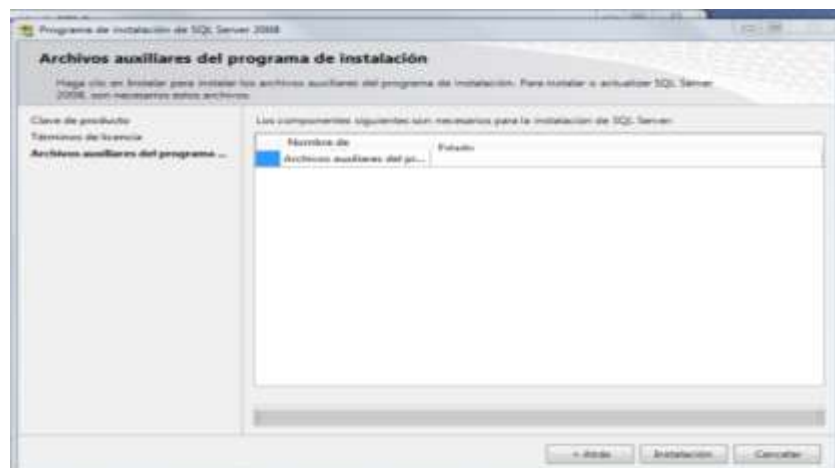


Figura 2.3.4.6. Archivos Auxiliares de la instalación.

La figura 2.3.4.6. Muestra la instalación componentes y archivos de soporte para Sql Server 2008.



Figura 2.3.4.7. Reglas auxiliares del programa de instalación.

La Figura 2.3.4.7. Muestra el reporte de requisitos para la instalación de Sql Server 2008.

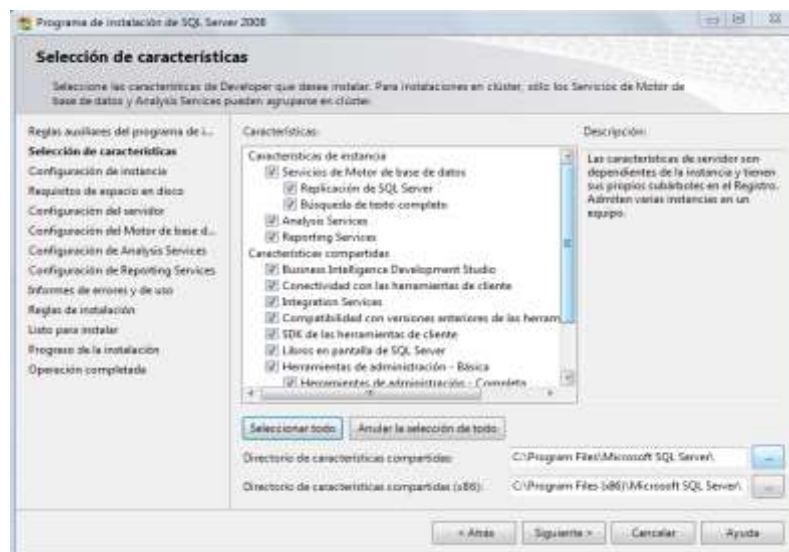


Figura 2.3.4.8. Selección de características.

La figura 2.3.4.8. Muestra la interfaz en la que se seleccionan las características de instalación a utilizar.

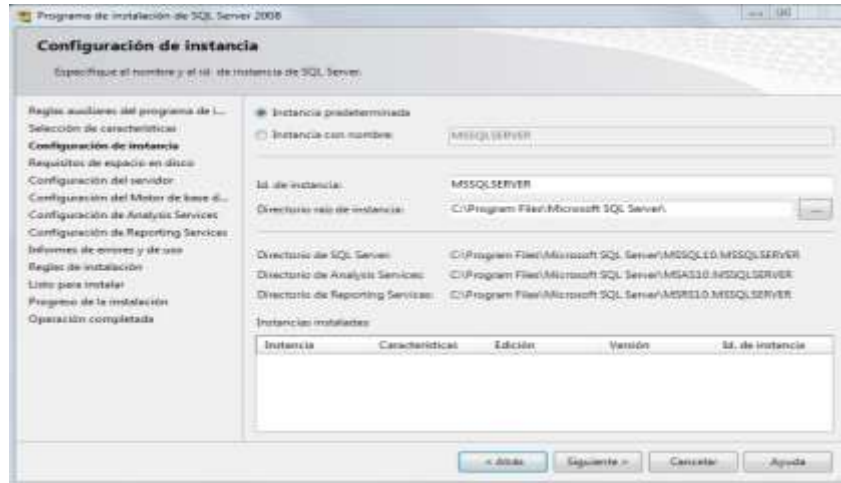


Figura 2.3.4.9. Configuración de la instancia.

La figura 2.3.4.9. Muestra la interfaz de selección y configuración de la instancia.

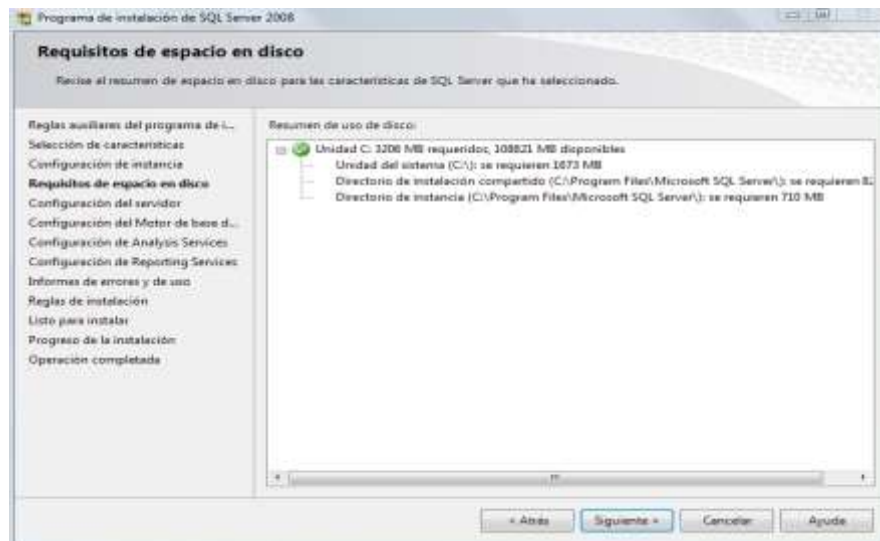


Figura 2.3.4.10. Requisitos de espacio en disco.

La figura 2.3.4.10. Calcula el requisito de espacio necesario para la instalación del producto.

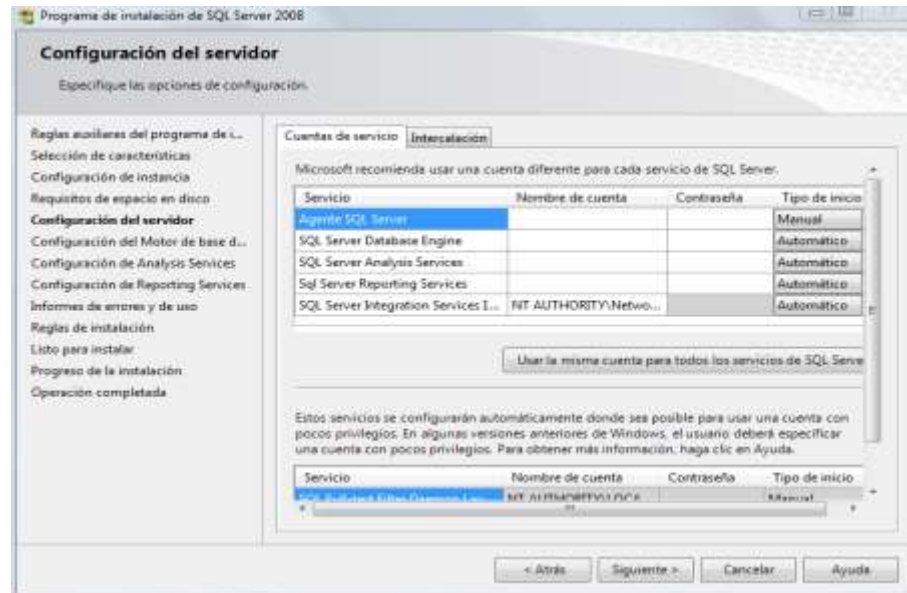


Figura 2.3.4.11. Configuración del servidor.

Figura 2.3.4.11. Muestra la selección del nombre del servidor a utilizar, en este caso se seleccionara el botón “usar la misma cuenta para todos los servicios...”

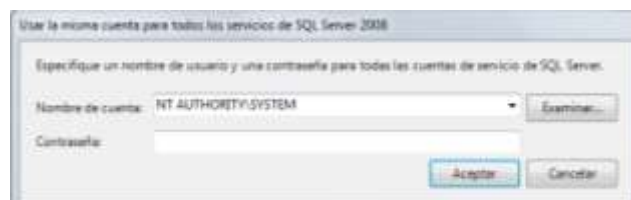


Figura 2.3.4.11.1 Usar la misma cuenta.

Figura 2.3.4.11.1 Muestra la selección de la cuenta para todos los servicios el cual se escogerá AUTHORITY/SYSTEM.

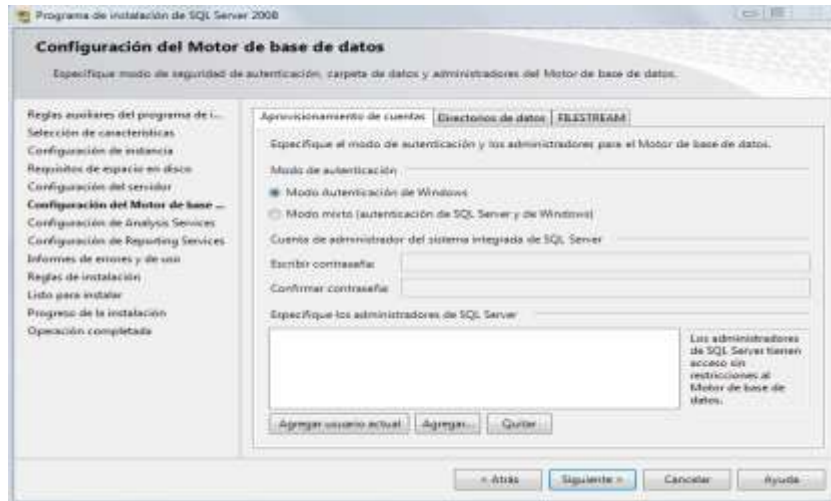


Figura 2.3.4.12. Configuración del motor de base de datos.

La figura 2.3.4.12. Muestra la selección de la autenticación de la base de datos el cual puede ser escogido como “autenticación de windows” o “modo mixto” y se agrega el usuario actual.

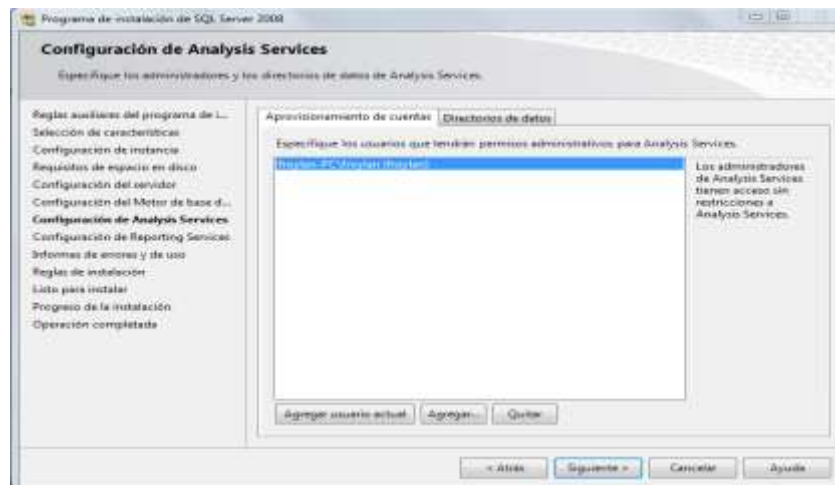


Figura 2.3.4.13. Configuración de analysis services.

La figura 2.3.4.13. Muestra la especificación de las directivas de Analysis Services, seleccionar “usuario actual”.



Figura 2.3.4.14. Configuración de reporting services.

La figura 2.3.4.14. Muestra la interfaz de selección del modo de configuración de reporting services.



Figura 2.3.4.15. Informe de errores y de uso.

La figura 2.3.4.15. Muestra la interfaz de selección de envío de errores y uso a Microsoft.

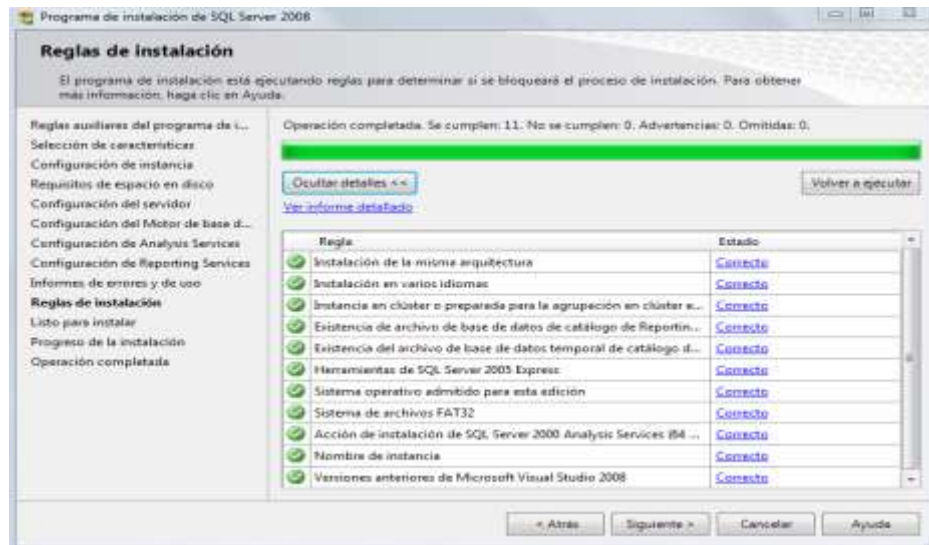


Figura 2.3.4.16. Reglas de instalación.

La figura 2.3.4.16. Muestra si las reglas de instalación de Sql Server 2008 están correctamente configuradas y si se cumplen los requisitos.

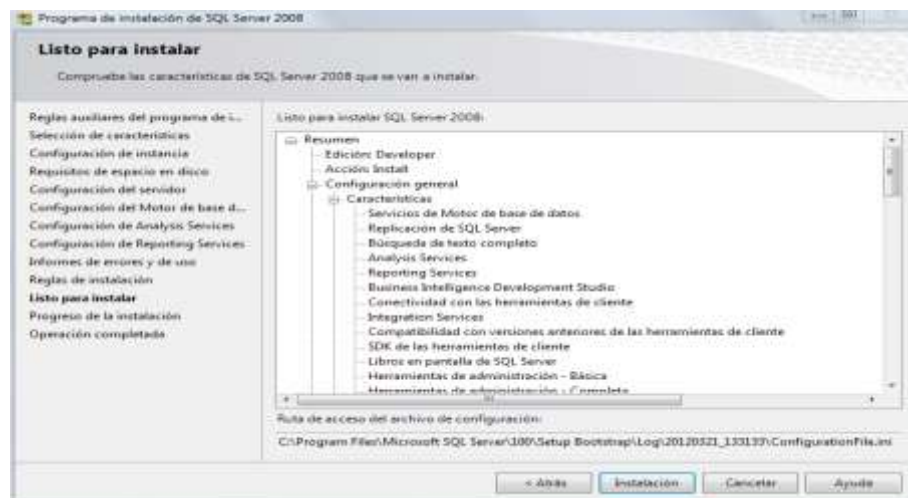


Figura 2.3.4.17. Listo para instalar.

La figura 2.3.4.17. Muestra el reporte de verificación si todo esta correctamente configurado.

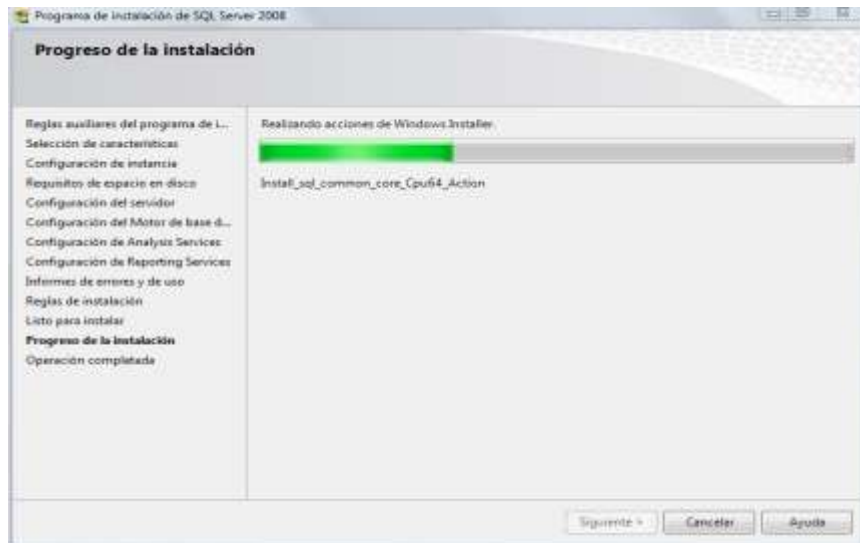


Figura 2.3.4.18. Proceso de instalación.

Figura 2.3.4.18. Muestra el proceso de instalación y configuración de todos los componentes.

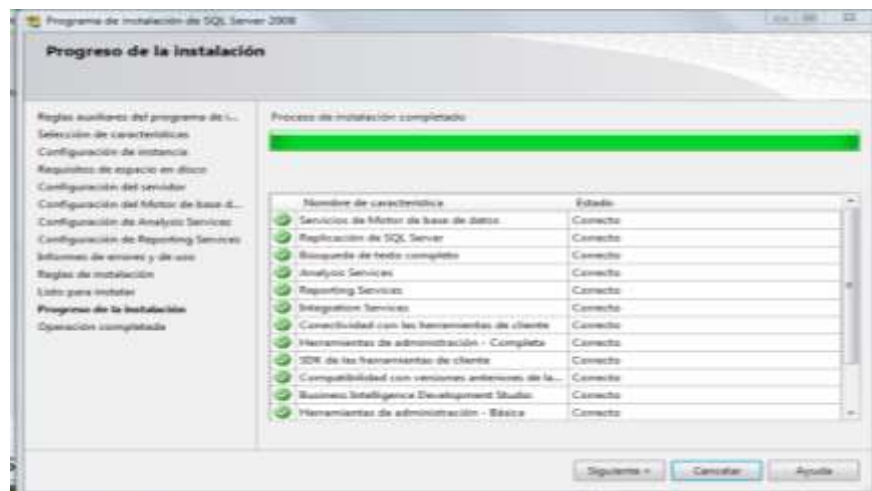


Figura 2.3.4.19. Proceso de instalación finalizado.

Figura 2.3.4.19. Muestra si los componente de instalación fueron instalados y configurados exitosamente.



Figura 2.3.4.20. Operación completada.

La figura 2.3.4.20. Muestra el final de la instalación exitosa y se selecciona cerrar.

CAPÍTULO III: DISEÑO DE LA SOLUCIÓN.

El Cd Interactivo estará distribuido de la mejor manera posible para que el estudiante se le facilite entender e interpretar en qué consiste, con sus diferentes temáticas para darle un mejor uso y entendimiento a los temas ya desarrollados y así el estudiante logre interactuar entre éste y su contenido, el estudiante tendrá la facilidad de tener el cd-interactivo, introducirlo a una pc y ejecutarlo ya sea de modo automático o ejecutándolo desde el panel de mi pc, ya en el momento de interactuar el estudiante tendrá acceso a las diferentes interfaces que muestran cada unidad que se desarrolla en la asignatura de Desarrollo de Sistemas Informáticos Web II con sus temas específicos, así como también podrá interactuar con videos donde se ha realizado ejemplos prácticos e instalación de los requerimientos de sistemas para un mejor entendimiento de la asignatura.

3.1. Diseño del Material Didáctico.

El CD-Interactivo está comprendido con las diferentes interfaces que muestran cada metodología y temáticas a desarrollar, por tal razón es de vital importancia tener presente que cada desarrollo de solución, tiene el objetivo primordial de mejorar el aprendizaje de los estudiantes y así mismo que aprendan a interactuar con los diferentes temas de desarrollo, entre las cuales se muestran las diferentes pantallas que tienen interactividad para que el estudiante se

familiarice y vaya mirando videos con ejercicios ya resueltos como también guías de ejercicios prácticos para que los puedan resolver...:



Figura 3.1.1. Pantalla principal.

En la figura 3.1.1. Muestra la Pantalla inicial del CD interactivo, la cual tendrá diferentes opciones de navegación como: Configuración del Entorno de Desarrollo, Desarrollo de Sistemas de información Transaccionales, Desarrollo de Membership, Desarrollo de Sistemas de Información usando Linq, Web Service y Ajax, también el manual de usuario y ejercicios propuestos.



Figura 3.1.2. Configuración e instalación del software.

En la figura 3.1.2. Se muestran las opciones de navegación: Instalación de IIS, Instalación de SQL Server 2008 y Microsoft Visual Studio 2010; su contenido está elaborado en videos de instalación de los entornos de desarrollo.

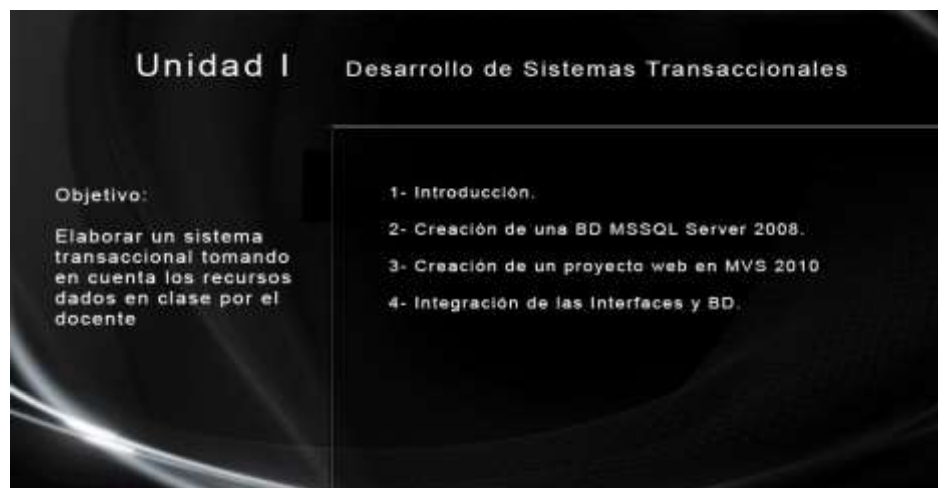


Figura 3.1.3. Desarrollo de sistemas transaccionales.

En la figura 3.1.3. Se muestra la pantalla con las diferentes opciones de navegación entre ellas están: Introducción de la unidad, Creación de una Base

de Datos SQL Server, Creación de un Proyecto Web en Microsoft Visual Studio 2010 y la integración entre ambos.

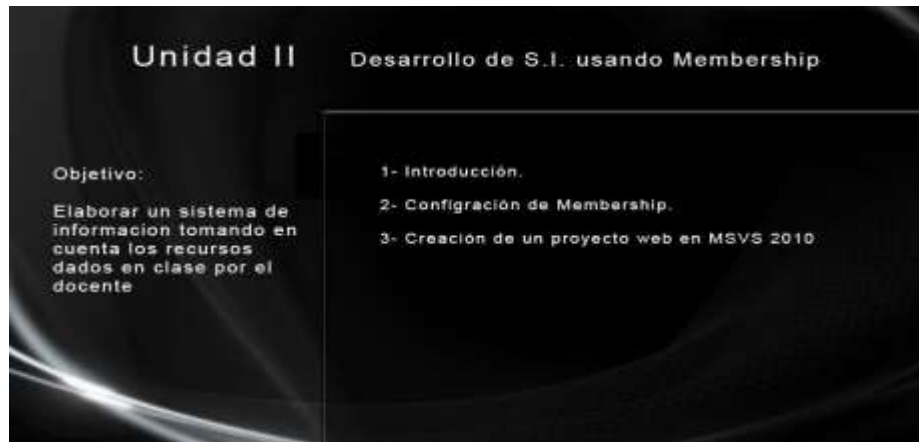


Figura 3.1.4. Desarrollo de sistemas usando Membership.

En la figura 3.1.4. Se muestra la pantalla que contiene dos opciones de desarrollo de Sistemas Informáticos Web usando la seguridad integrada de Membership, las opciones de navegación son: Introducción de la unidad, Configuración de Membership y Creación de un Proyecto Web utilizando la seguridad integrada.



Figura 3.1.5. Desarrollo de S.I usando tecnología LinQ.

La figura 3.1.5. Muestra en la pantalla las opciones de navegación: Introducción de la unidad, Creación de procedimientos almacenados, un proyecto web y la integración entre ambas tecnologías.



Figura 3.1.6. Desarrollo de sistemas de información usando clases y Web Service.

La figura 3.1.6. Muestra las pantallas de navegación, seleccionando las opciones que se necesitan conocer: Introducción de la unidad, Crear un Webservice, Consumir un Webservice, Crear un Webservice con conexión a BD con ADO.Net, Consumir un Webservice con conexión a BD, Descargar proyecto Webservice y Descargar proyecto cliente.

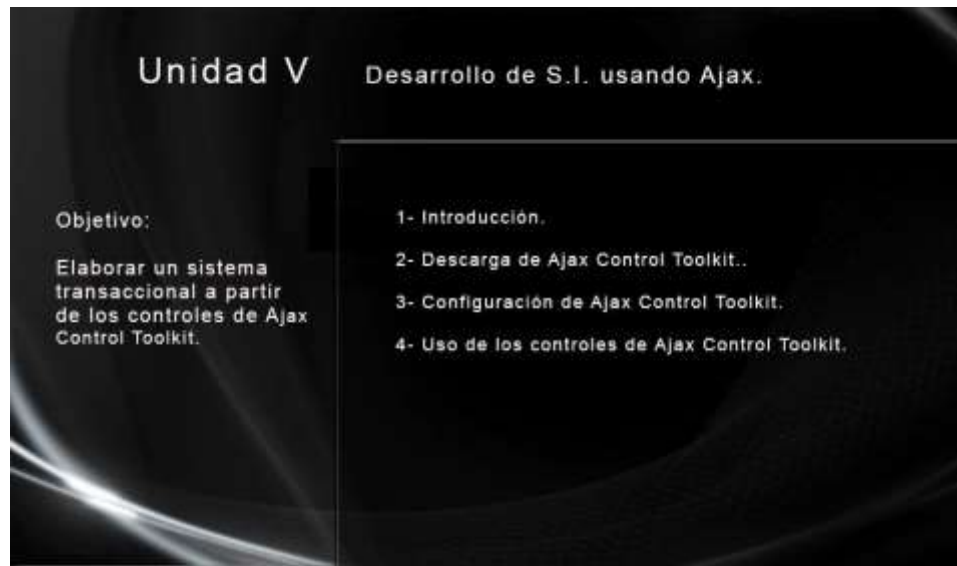


Figura 3.1.7, Desarrollo de sistemas de información usando Ajax.

La figura 3.1.7. Muestra en la pantalla las diferentes opciones de navegación entre las cuales se puede seleccionar dependiendo lo que se desea realizar: Introducción de la unidad, Descarga de Ajax Control Toolkit, Instalación de Ajax Control Toolkit y el uso de los controles de Ajax.



Figura 3.1.8. Menú de Navegación del Cd Interactivo.

La figura 3.1.8. Muestra el menú de opciones de navegación que dirigen hacia las diferentes pantallas que consta el Cd-Interactivo.

3.2. Diseño de Ejemplos Prácticos.

Objetivo: Facilitar al estudiante los diferentes temas y desarrollos que conlleva la asignatura para que pueda entender y practicar los conocimientos adquiridos en las diferentes materias ya cursadas.

1. UNIDAD I: Desarrollo de Sistemas Informáticos Transaccionales.

- 1.1. Un cliente transfiere dinero de una cuenta a otra cuenta dentro de un mismo banco; la cantidad de dinero que se descuenta de la cuenta emisora debe ser igual a la que se suma en la cuenta receptora. De no ser así, la acción (transacción) no se realiza; esto es un ejemplo claro de un sistema transaccional ya que si la operación o la acción no se realiza no estaría consumiendo o realizando nada.
- 1.2. Un cliente está haciendo la reserva de un asiento en un vuelo, dicho asiento debe ser bloqueado temporalmente hasta que se concrete la transacción, porque otro cliente podría estar queriendo reservar el mismo asiento en el mismo momento, como su nombre lo dice si un cliente quiere hacer la transacción y otro también debe bloquearse temporalmente, si esto no fuera así podría existir deficiencia en el negocio y no sería un sistema transaccional.
- 1.3. El sistema transaccional de una aerolínea debe aceptar reservas de múltiples agencias de viajes. Cada reserva debe contener los mismos datos obligatorios, con determinadas características.

1.4. Un sistema transaccional claro es el de una venta de accesorios para computadoras a través de la Web, ya que el o los usuarios conectados en ese mismo momento podrían adquirir el mismo producto con determinadas características, en ese momento se debe bloquear para que no afecte a ningún usuario y pueda adquirir el producto.

1.5. Otro ejemplo de sistemas transaccionales es de la compañía Ebay que brinda el servicio a los usuarios con la venta de productos, por lo general son productos usados a través de lugares especializados según su rango.

1.6. Creación de Base de datos:

Para el ejemplo, se creará una base de datos llamada “Estudiantes”.

- ✓ Se abre SQL Managment Studio

(Inicio – Todos los programas – Microsoft SQL Server 2008 – SQL Server Management Studio).

- ✓ Al momento de conectar al servidor, se cambia el nombre del servidor (a local) y se selecciona conectar.

- ✓ Luego en la barra del explorador de objetos se da click derecho donde dice BASES DE DATOS.

- ✓ Continuamente se selecciona NUEVA BASE DE DATOS.

- ✓ Posteriormente se desplegará una ventana en la que se ingresara el nombre de la base de datos que en este caso será “Estudiantes”, y se presiona el botón aceptar.

1.7. Creación de base de datos con T-SQL:

- ✓ Para continuar con los ejemplos se realizan siguiendo los pasos anteriores para iniciar Microsoft SQL Server 2008, partiendo ya de la pantalla de este se hace lo siguiente:
- ✓ Se selecciona la opción Nueva consulta, esto desplegará una ventana en blanco la cual servirá para digitar lo que desea realizar o también dando click derecho en el cuadro de explorador de objetos y luego se selecciona en nueva consulta.
- ✓ Posteriormente se escribe en la nueva consulta la sentencia “create database Inventario” o se puede pegar el script que ya ha sido creado o haya sido proporcionado.
- ✓ Luego se selecciona ejecutar o en la tecla F5.
- ✓ Automáticamente la base de datos ha sido creada.

1.8. Creación de tablas:

Tabla alumnos:

- ✓ Se da click derecho en la opción de tablas que se encuentra en las subcarpetas de la base de datos ya creada llamada “Estudiantes” y posteriormente click donde dice “nueva tabla...”

- ✓ Se despliega una ventana en la que se insertará los campos necesarios para la tabla llamada “alumnos”: identificación, pri_nom, seg_nom, pri_ape, seg_ape, genero, teléfono, dirección; los tipos de datos son: int, varchar (50), varchar (50), varchar (50), varchar (50), char (1), int y varchar (100) respectivamente.
- ✓ Luego se da click derecho en el campo de identificación y se desplegará otra ventana, se le asigna “establecer clave principal”.
- ✓ Se selecciona guardar y se digita el nombre de la tabla el cual será “alumnos”.

Tabla Cursos:

- ✓ Se realizan los primeros pasos de ejemplo anterior, con la diferencia que la tabla contara con dos campos, estos serán: id y nombre, los tipos de datos son: int y varchar (50) respectivamente.
- ✓ Luego se establece el id como clave principal y se guarda con el nombre de “cursos”.

Tabla matriculas:

- ✓ Se realizan siempre los primeros pasos de ejemplo anterior, con la diferencia de la tabla matricula que contara con dos campos, estos serán: id, id estudiante, id_curso, id_matricula, con los respectivos tipos de datos: int, int, int y datetime.

- ✓ Luego se establece el id como clave principal y se guarda el archivo con el nombre de “matricula”.

1.9. Creación de tablas con T-SQL:

- ✓ Como primer punto para crear la tabla en la base de datos que ya ha sido creada se realiza lo siguiente:
- ✓ Seleccionar “nueva consulta” para que se despliegue la ventana para digitar o pegar algún script.
- ✓ Luego digitar en la ventana de consulta “use Inventario” (esto indica que voy a usar la base de datos Inventario ya que pueden existir muchas más bases de datos) y en la siguiente línea la palabra “go”, luego seleccionar ejecutar o presionar la tecla F5 o también en la parte superior se encuentra un menú con un icono que por default dice “master” se da click en esa opción y desplegará las opciones de base de datos que han sido creadas y se selecciona la base de datos “Inventario”.
- ✓ Se procede a crear la tabla digitando en la siguiente línea de la ventana, “create table Modulo”.
- ✓ Posteriormente se crearán los campos que contiene la tabla, digitando cada uno de ellos con sus características de la siguiente manera: Create table Modulo (CodigoModulo int primary key not null, Nombre varchar (100));

- ✓ Luego se selecciona lo digitado y seleccionar ejecutar o presionar la tecla F5.

1.10. Otras tablas referenciadas en la base de datos Inventario:

Para realizar el siguiente ejemplo es necesario contar con el script de la base de datos para acceder a este, se debe de descargar desde el cd- interactivo en la opción **UNIDAD I: Sistemas Transaccionales – Ejemplo de cómo crear Base de Datos**; luego realizar los pasos anteriormente mencionados en el numeral 1.7 donde dice pegar script proporcionado.

1.11. Creación de diagrama de base de datos:

- ✓ Para acceder a crear el diagrama de la base de datos se siguen los siguientes pasos:
- ✓ Se selecciona click derecho en la subcarpeta de “Diagramas de bases de datos” de la base de datos “Estudiantes”.
- ✓ Al efectuar esta acción (por ser la primera práctica en Microsoft SQL Server 2008), se despliega una ventana en donde hace una pregunta a la cual se responderá con “sí”.
- ✓ Posteriormente se selecciona click derecho sobre la subcarpeta de “Diagramas de bases de datos” de la base de datos “Estudiantes” y se selecciona la opción “Nuevo diagrama de bases de datos”.

- ✓ Luego se despliega una ventana en la cual se seleccionan las tablas que se han creado en este caso alumnos, cursos y matricula, posteriormente seleccionar agregar y luego en la opción cerrar.
- ✓ Al realizar estos pasos se despliegan las tablas creadas estas se organizan de la mejor manera posible para que sea más entendible y encontrarle la relación.

Este ejemplo se puede ver también en los videos que han sido creados en el cd – interactivo en la sección de **UNIDAD I: Sistemas de Información Transaccionales**

1.12. Crear un Proyecto en Microsoft Visual Studio 2010:

- ✓ Se inicia Microsoft Visual Studio 2010, donde se selecciona la opción nuevo proyecto, ubicado en la parte izquierda de la ventana principal.
- ✓ Posteriormente se le asigna el nombre con el que se guardará: en este caso se llamara “EjemploConexion” y la ubicación donde se encontrará, si no se le llegase a asignar un nombre o ubicación Microsoft Visual Studio lo guardara por default en la carpeta de “projects” y dicha carpeta está ubicada en el lugar donde se encuentra instalado Microsoft Visual Studio 2010.
- ✓ En esa parte muestra un menú de opciones de proyectos que se pueden realizar en la aplicación con diferentes lenguajes, en este

caso se selecciona “C#” y muestra un submenú de opciones con las plantillas para realizar diferentes proyectos, se selecciona en file nuevo sitio web y luego se selecciona aceptar.

- ✓ Realizado este paso se muestra la ventana en la que se crea el sitio web y en la parte izquierda de abajo de la ventana se selecciona la opción en diseño, muestra una interfaz en blanco donde se muestra el menú de herramientas con diferentes controles que se pueden usar para crear aplicaciones web.
- ✓ Para este ejemplo se creara el formulario que se muestra a continuación:



The screenshot shows a web application window titled "Controlador Base de Datos". It contains a form with the following fields and controls:

- Labels: CARNET, NOMBRES, APELLIDOS, EMAIL, TELEFONO, DIRECCION.
- Buttons: MOSTRAR DATOS (above the email field), AGREGAR NUEVO (below the address field), GUARDAR (below the address field), BUSCAR (next to the DUI field), and EDITAR (below a large grey rectangular area).
- Input fields: Empty text boxes for CARNET, NOMBRES, APELLIDOS, EMAIL, TELEFONO, DIRECCION, and DUI.

Otro ejemplo similar sobre cómo crear un proyecto web en Microsoft Visual Studio 2010 puede observarse en el cd – interactivo en la sección de la **UNIDAD I: Sistemas Información Transaccionales**.

1.13. Conexión entre SQL Server 2008 y Microsoft Visual Studio 2010.

Para poder ver un ejemplo de cómo crear la conexión entre SQL Server 2008 y Microsoft Visual Studio 2010 diríjase al video del cd-interactivo en la opción de **UNIDAD I: Sistemas de Información Transaccionales – Ejemplo de Sistema Transaccional**.

2. UNIDAD II: Desarrollo de Sistemas Informáticos con Membership

2.1. Base de datos con seguridad integrada (Membership).

- ✓ Para obtener una base de datos con seguridad integrada es necesario realizarlo desde el asistente de SQL Server 2008, este ejecutable se encuentra en una carpeta determinada que permite el acceso para la seguridad de Membership; el ejecutable se llama aspnet_regsql.exe, y para tener acceso a él se debe ir a la siguiente dirección:

C:\windows\microsoft.net\frameworks\v4.0.3935\

- ✓ Luego de realizar esta opción aparece la ventana del asistente en el cual se selecciona los siguientes pasos en:
- ✓ Configuración de Aplicaciones.
- ✓ Colocar en el Servidor de la Base de Datos (local).
- ✓ Se selecciona la base de datos a la cual se le aplicara Membership en este caso “Inventario”.
- ✓ Terminar de ejecutar el asistente.
- ✓ Posteriormente ya la seguridad de Membership ha sido creada en la base de datos Inventario; para probar que se creó se accede a

Management Studio y entrar con el nombre de servidor (como local).

- ✓ Expandir la base de datos Inventario y verificar las tablas creadas y procedimientos almacenados que han sido creados por medio de Membership.

2.2. Integración de la base de datos con Membership y Microsoft Visual Studio 2010.

- ✓ Para realizar la integración de la base de datos Membership creada anteriormente se accede a la aplicación de Microsoft Visual Studio 2010.
- ✓ Crear un nuevo Proyecto Web ASP.NET llamado Prueba (proyecto vacío).
- ✓ Modificar el connection String del archivo web.config como= "Data Source=(local); Initial Catalog=Inventario; Integrated Security=true".
- ✓ Ejecutar la aplicación con F5; ante esta opción el programa se ejecutara mostrando una interfaz a la cual se puede acceder ya sea creando un usuario o en un usuario ya creado, para este caso habría que crearlo porque sería la primer ejecución.

Para visualizar en un video como se realiza estos procedimientos más detallados refiérase al Cd – Interactivo en la Sección **UNIDAD II: Desarrollo de Sistemas Informáticos con Membership.**

3. **UNIDAD III:** Desarrollo de Sistemas Informáticos con tecnología Linq

3.1. **Crear procedimientos almacenados:**

- ✓ El siguiente ejemplo muestra un procedimiento almacenado, denominado sp_addCliente que inserta un registro en la tabla "CLIENTES".

```
CREATE PROCEDURE sp_addCliente  
  
@nombre varchar (100),  
  
@apellido1 varchar (100),  
  
@apellido2 varchar (100),  
  
@nit varchar (20),  
  
@fecha_Nacimiento datetime  
  
AS  
  
INSERT INTO CLIENTES (nombre, apellido1, apellido2, nit,  
fecha_Nacimiento)  
  
VALUES (@nombre, @apellido1, @apellido2, @nit,  
@fecha_Nacimiento)
```

- ✓ Para la ejecución de un procedimiento almacenado se debe utilizar la sentencia EXEC. Cuando la ejecución del procedimiento almacenado es la primera instrucción, podemos omitir el uso de EXEC.

- ✓ Siempre es deseable que las instrucciones del procedure estén dentro de un bloque TRY CATCH y controlados por una transacción por tal razón el siguiente ejemplo lo muestra.

```
ALTER PROCEDURE sp_addCliente
```

```
@nombre varchar (100),
```

```
@apellido1 varchar (100),
```

```
@apellido2 varchar (100),
```

```
@nit varchar (20),
```

```
@fecha_Nacimiento datetime
```

```
AS
```

```
BEGIN
```

```
TRY
```

```
    BEGIN TRAN
```

```
    INSERT INTO CLIENTES (nombre, apellido1, apellido2, nit,  
    fecha_Nacimiento)
```

```
    VALUES (@nombre, @apellido1, @apellido2, @nit,  
    @fecha_Nacimiento)
```

```
    COMMIT
```

```
END
```

```
TRY
```

```
BEGIN
```

```
CATCH
```

```
ROLLBACK  
  
PRINT ERROR_MESSAGE ()  
  
END  
  
CATCH
```

Para conocer más sobre cómo se elaboran los procedimientos almacenados de agregar, borrar y eliminar en SQL Server 2008 puede consultarlo en el siguiente ejemplo mostrado en el Cd – Interactivo en la opción de **UNIDAD III: Desarrollo de Sistemas Informáticos con tecnología Linq.**

También puede consultar en la sección de la **UNIDAD III: Desarrollo de Sistemas Informáticos con tecnología Linq**, en la que se le explicara cómo realizar este tipo de conexiones entre SQL Server y Microsoft Visual Studio 2010.

4. UNIDAD IV: Desarrollo de Sistemas Informáticos utilizando Clases y Web Services.

4.1. Como crear un Web Service en Microsoft Visual Studio 2010.

- ✓ Ejecutar Microsoft Visual Studio 2010.
- ✓ Cuando muestre la pantalla de bienvenida, seleccionar en Archivo—► nuevo —► Sitio Web.
- ✓ Seleccionar lenguaje de programación C#.

- ✓ Seleccionar la opción de aplicación ASP.net Web Service Application.
- ✓ Nombrar al servicio web “servicioWebPrueba”.
- ✓ Seleccionar Aceptar.
- ✓ Posteriormente escribir un método o varios métodos en este caso el siguiente método el cual retornada la edad que es un método sencillo y práctico.

[WebMethod]

```
public string edad (string edad)
{
    return ("mi edad es: " + edad);
}
```

- ✓ Luego de haber digitado el método Presionar F5 o ejecutar.
- ✓ Visualizar en el navegador el nombre del servicio web y el nombre del método.
- ✓ Copiar el url.

Para tener un ejemplo más claro sobre la creación de un Web Service de Microsoft Visual Studio 2010 consulte la sección de la **UNIDAD IV: Desarrollo de Sistemas Informáticos utilizando Clases y Web Services**, del Cd – Interactivo, ahí encontrara un video con un ejemplo ya desarrollado.

4.2. Como consumir un Web Service.

Para consumir el Web Service creado anteriormente se deben realizar los siguientes pasos:

- ✓ Abrir una nueva ventana de Microsoft Visual Studio 2010.
- ✓ Archivo → nuevo → proyecto
- ✓ Seleccionar lenguaje de programación C#.
- ✓ Seleccionar sitio web ASP.NET.
- ✓ Nombrar al sitio web como “aplicacionCliente”.
- ✓ En el formulario default.aspx crear dos textbox y un botón desde el panel.
- ✓ En el explorador de soluciones click derecho sobre el directorio “references”.
- ✓ Agregar una referencia web.
- ✓ Luego en textbox “url” pegar el url del WebService anteriormente creado y seleccionar Enter.
- ✓ Asignarle el nombre al campo de “nombre de la referencia web” con el nombre de “WebServiceCliente”
- ✓ Seleccionar agregar referencia y cuando aparezca la ventana, pegar el siguiente código en el evento del botón.

```
WebServiceCliente.Service servicio = new
WebServiceCliente.Service ();

TextBox2.Text = servicio.edad (TextBox1.Text);
```

- ✓ Ejecutar el sitio web presionando F5 o ejecutar.

Otro ejemplo de cómo consumir un Web Service en Microsoft Visual Studio 2010 se puede verificar en el video que ha sido creado en el Cd – Interactivo en la sección de **UNIDAD IV: Desarrollo de Sistemas Informáticos utilizando Clases y Web Services.**

5. **UNIDAD V:** Desarrollo de Sistemas Informáticos utilizando Ajax.

Visual Studio cuenta con una gama de controles en la cual se pueden crear muchas aplicaciones y sistemas de gran interés que facilita en buen funcionamiento de estos y para acceder a los controles de Ajax es necesario descargarlos para que se agreguen al panel.

5.1. Descarga de Controles Ajax.

- ✓ Como primer punto se debe ingresar en la siguiente dirección;
<http://ajaxcontroltoolkit.codeplex.com/>
- ✓ Seleccionar la opción Descargar Ajax Control Toolkit.
- ✓ Luego cuando esto haya finalizado se busca en la carpeta que ha sido descargado y se descomprime el contenido del archivo en una carpeta.
- ✓ Ejecutar Visual Studio 2010.
- ✓ Archivo –► nuevo –► sitio web.
- ✓ Seleccionar lenguaje de programación C#.
- ✓ Seleccionar sitio web ASP.NET.
- ✓ En el cuadro de herramientas presionar click derecho, y seleccionar agregar nueva ficha.

- ✓ Nombrar la ficha como Ajax Control Toolkit.
- ✓ Seleccionar click derecho sobre la ficha Ajax Control Toolkit y posteriormente “elegir elementos”.
- ✓ En la ventana emergente seleccionar el botón examinar y buscar la carpeta donde se descomprimió la carpeta de Ajax Control Toolkit.
- ✓ Seleccionar la AjaxControlToolkit.dll y luego seleccionar Aceptar.

5.2. Creación de Controles de Ajax Control Toolkit.

- Agregar el control CalendarExtender.
- ✓ Realizar los pasos anteriormente mencionados en el literal 5.1 y luego dirigirse al formulario de la masterpage y agregar el control toolkitscriptmanager.
- ✓ En la masterpage agregar el control textbox.
- ✓ Sobre el control textbox arrastrar el control CalendarExtender o en las tareas del textbox seleccionar la opción de agregar extensor y seleccionar el control CalendarExtender.
- ✓ Luego ejecutar y al momento que la interfaz se muestra al seleccionar la casilla se mostrara un calendario con la fecha y ahí se selecciona la que se desee.
 - Agregar el control ColorpickerExtender.

- ✓ Realizar los pasos anteriormente mencionados en el literal 5.1 y luego dirigirse al formulario de la masterpage y agregar el control `toolkitscriptmanager`.
- ✓ En la masterpage agregar el control `textbox`.
- ✓ Sobre el control `textbox` arrastrar el control `ColorpickerExtender` o en las tareas del `textbox` seleccionar la opción de agregar extensor y seleccionar el control `ColorpickerExtender`.
- ✓ Al realizar esta opción seleccionar ejecutar o en la tecla F5 y mostrar el proyecto en ejecución para que se pueda verificar que seleccionar la casilla que muestra un cuadro de colores al cual se puede seleccionar el que se desee.

3.3. Diseño de ejercicios Propuestos.

Objetivo: Profundizar los conocimientos de los estudiantes en el marco de desarrollo de los diferentes métodos de realizar bases de datos y la relación entre los sitios web.

3.4.1. UNIDAD I:

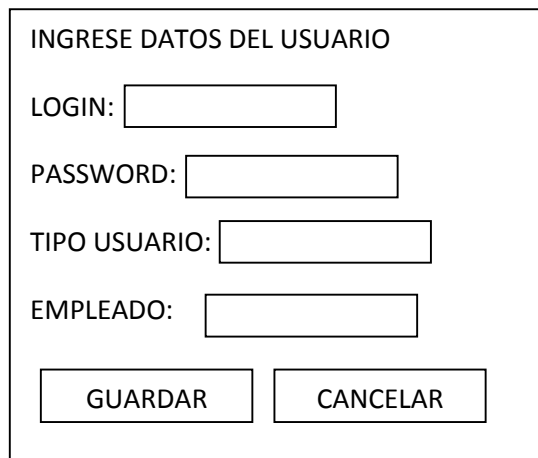
- Elabore la base de datos con sus tablas y su diagrama para respectivo el literal 1.1 de la sección 3.3. Diseño de Ejemplos Prácticos.
- Realice los procedimientos de agregar, modificar y eliminar del literal 1.1 de la sección 3.3. Diseño de Ejemplos Prácticos.

3.4.2. UNIDAD II:

- ✓ Leer el siguiente enunciado y elaborar la base de datos con sus diferentes tablas.
 - ✓ Una base de datos para una pequeña empresa debe contener información acerca de clientes, artículos y pedidos. Hasta el momento se registran los siguientes datos en documentos varios:
 - ✓ Para cada cliente: Número de cliente (único), Direcciones de envío (varias por cliente), Saldo, Límite de crédito (depende del cliente, pero en ningún caso debe superar los \$3.000.000), Descuento.
 - ✓ Para cada artículo: Número de artículo (único), fábricas que lo distribuyen, existencias de ese artículo en cada fábrica, descripción del artículo.
 - ✓ Para cada pedido: Cada pedido tiene una cabecera y el cuerpo del pedido. La cabecera está formada por el número de cliente, dirección de envío y fecha del pedido. El cuerpo del pedido son varias líneas, en cada línea se especifican el número del artículo pedido y la cantidad.
 - ✓ Además, se ha determinado que se debe almacenar la información de las fábricas. Sin embargo, dado el uso de distribuidores, se usará: Número de la fábrica (único) y Teléfono de contacto. Y se desean ver cuántos artículos (en total) provee la fábrica.

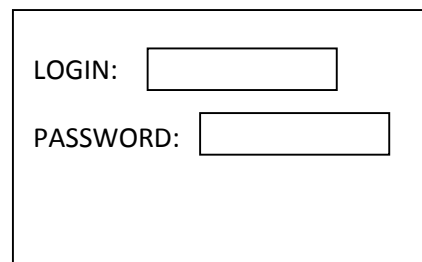
Nota: Una dirección se entenderá como N°, Calle, Colonia y Ciudad.

- ✓ Elabore los siguientes formularios en Visual Studio 2010 como se muestran a continuación.
- ✓ Cree los formularios USUARIO ADD.ASPX e INGRESAR.ASPX



Formulario USUARIO ADD.ASPX. El formulario tiene un título "INGRESE DATOS DEL USUARIO". Debajo del título hay cuatro campos de texto etiquetados como "LOGIN:", "PASSWORD:", "TIPO USUARIO:" y "EMPLEADO:". En la parte inferior del formulario hay dos botones: "GUARDAR" y "CANCELAR".

USUARIO ADD.ASPX



Formulario INGRESAR.aspx. El formulario tiene dos campos de texto etiquetados como "LOGIN:" y "PASSWORD:".

INGRESAR.aspx

- ✓ Cree la base de datos Facturación.
- ✓ Elaborar las tablas Productos, Ventas, Reportes, Empleado, Usuario.
- ✓ Crear el procedimiento almacenado de agregar usuario para el formulario USUARIO ADD.ASPX e INGRESAR.ASPX.
- ✓ Cree una base de datos llamada REGISTRO_ACADEMICO en la cual se llevara el registro de los grupos, inscripciones, alumnos, evaluaciones y notas.

- ✓ Realice las modificaciones necesarias al siguiente script para la creación de cada una de las tablas en SQL Server 2010, las tablas a crear son ALUMNO,GRUPO, EVALUACION, INSCRIPCION, NOTA

create table alumno

(carnetchar (8) not null,

nombre1 char (15) not null,

nombre2 char (15) null,

apellido1 char (15) not null,

apellido2 char (15) null,

sexo char (1) not null,

constraintpk_carnet primary key (carnet),

constraintck_sexo check (sexo like '[MF]'))

create table grupo

(id_grupo char(4) not null,

Descripcionchar (60) not null,

Inscritosint,

Constraintpk_id_grupo primary key (id_grupo))

create table evaluacion

(id_evaluacion tinyint not null,

descripcion char (60) not null,

constraint pk_id_evaluacion primary key (id_evaluacion))

create table inscripcion

(id_grupo char (4) not null,

carnet char (8) not null,

constraint pk_inscripcion primary key (id_grupo, carnet),

constraint fk_id_grupo foreign key (id_grupo) references grupo (id_grupo),

constraint fk_carnet foreign key (carnet) references alumno (carnet))

create table nota

(id_grupo char (4) not null,

carnet char (8) not null,

id_evaluacion tinyint not null,

nota decimal null,

constraint fk_nota foreign key (id_grupo, carnet)

```

referencesinscripcion (id_grupo, carnet),

constraintu_notaunique (id_grupo,carnet,id_evaluacion),

constraintfk_id_evaluacion foreign key (id_evaluacion)

references evaluación (id_evaluacion))

```

✓ Ingrese los siguientes registros a las tablas ALUMNO, GRUPO y EVALUACION

Insertinto alumno values ('mb960240','Kelin', null, 'Chávez','Castro','f')

Insertinto alumno values ('om960201','Froylan', null,'Balmoris','Méndez','m')

Insert into alumno values ('mm960213','Santos', null, 'Chávez', null,'m')

insert into alumno values ('mm960217','Karen','Irina','Cabas', 'Berríos','f')

Insertinto grupo values ('BDD1','Grupo 01, Base de datos',0)

Insertinto grupo values ('BDD2','Grupo 02, Base de datos', 0)

Insertinto grupo values ('BDD3','Grupo 03, Base de datos', 0)

Insertinto evaluación values (1, 'Parcial práctico 1')

Insertinto evaluación values (2, 'Parcial práctico 2')

Insertinto evaluación values (3, 'Parcial práctico 3')

- ✓ Cree el sitio web Rent Car S.A. de C.V, mostrado en la imagen de la parte posterior, debe considerar el diseño, los colores y las imágenes.

Para llevar a cabo este literal debe tomar en cuenta lo siguiente:

- ✓ El sitio web debe ser creado en una carpeta llamada Autos y las imágenes deben encontrarse en una subcarpeta llamada imágenes.
- ✓ Haga uso de las imágenes proporcionadas.



Encabezado.jpg



Lateral.jpg



Chevy.jpg



Toyota.jpg

- Para la creación del sitio web debe utilizar tablas, puede hacer uso de tablas anidadas.
- Los colores utilizados son los siguientes:
 - Rojo → #cc0000
 - Gris → #a9a9a9

- Describa los pasos necesarios para subir el sitio web recién creado al servicio IIS:

3.4.3. UNIDAD III:

- ✓ Lleve a cabo la programación de un procedimiento almacenado que reciba como parámetros de entrada el código de un producto y un nuevo precio. Con estos parámetros de entrada debe llevar a cabo lo siguiente:
 - La modificación del precio en la tabla productos.
 - Debido a que el precio del producto ha cambiado es necesario actualizar el importe de todas las ventas ya efectuadas en la tabla ventas de ese producto.
 - Al cambiar el importe de las ventas también deben actualizarse las comisiones para cada vendedor, debido a que estas son calculadas en base a los importes de las ventas.

NOTA: Para la realización del literal anterior se sugiere un triggers de actualización en la tabla ventas que lleve a cabo la actualización del importe correspondiente.

- Un procedimiento almacenado que al momento de eliminar un vendedor elimine el registro asociado en la tabla comisiones.
- Un procedimiento almacenado que devuelva la edad de un vendedor, teniendo como parámetro de entrada el código del vendedor.
- ✓ Realice lo que se le pide a continuación:

- Ejecute el siguiente Script.

```
CREATE DATABASE bodega
```

```
GO
```

```
use bodega
```

```
GO
```

```
CREATE TABLE PRODUCTOS
```

```
(idprod char (7) PRIMARY KEY,
```

```
descripcion varchar(25),
```

```
existencias int)
```

```
CREATE TABLE PEDIDOS
```

```
(idpedido char (7),
```

```
Idprod char (7),
```

```
cantidad int
```

```
FOREIGN KEY (idprod) REFERENCES PRODUCTOS (idprod))
```

- Crear un procedimiento almacenado que ingrese los valores en la tabla PRODUCTOS, y deberá verificar que el código del producto no exista para poder insertarlo, en caso que el código del producto ya exista enviar un mensaje que diga ESTE PRODUCTO YA HA SIDO INGRESADO.
- Crear un procedimiento almacenado que permita realizar un pedido EN LA TABLA PEDIDOS, este procedimiento deberá verificar si el código del producto ingresado existe en la tabla

PRODUCTO, además si la cantidad a pedir del producto es mayor a la existencia del producto deberá enviar un mensaje que diga EXISTENCIA DEL PRODUCTO INSUFICIENTE, en caso que la cantidad a pedir sea menor o igual deberá modificar el valor de la existencia.

- ✓ Hacer un programa que haga un registro ya sea de notas, con Membership, Linq, procedimientos almacenados, implementando 10 controles Ajax y que estos sean verificables en su desarrollo.

3.4.4. UNIDAD IV:

- Crear y consumir un Web Service que retorne la edad a partir del año de nacimiento
- Crear y consumir un Web Service que retorne el promedio de un estudiante calculando 4 notas.
- Crear y consumir un Web Service que resuelva el siguiente ejercicio:
- Ingrese las horas y minutos que estuvo el vehículo en la playa de estacionamiento, se deberá visualizar el pago que hará el cliente sabiendo que por cada hora o fracción de la misma se cobra S/. 2.50.

3.4.5. UNIDAD V:

- Investigar el funcionamiento del control de passwordstrength y hacer un ejemplo de su uso.
- Explicar el funcionamiento del control confirmPasswordExtender y hacer un ejemplo de su uso.

- Explicar el funcionamiento del control DragPanelExtender y hacer un ejemplo de su uso.
- Investigar el funcionamiento del control de DropShadowExtender y hacer un ejemplo de su uso.
- Investigar el funcionamiento del control de CascadingDropDownny hacer un ejemplo de su uso.

RECOMENDACIONES

- ✓ El material didáctico es de gran importancia para la Universidad como para docentes y alumnos porque ayuda a desarrollar las temáticas de la asignatura por tal razón debe implementarse a corto plazo.
- ✓ El Cd-Interactivo debe ser proporcionado al estudiante para que se guíe de este y aprenda a interactuar entre los videos proporcionados y le facilite al docente implementar cada temática de la asignatura.
- ✓ La universidad será la encargada de tomar la decisión si el cd-interactivo será proporcionado al estudiante por algún costo extra o como será su distribución.

CONCLUSIONES

- ✓ El estudiante tendrá la facilidad de comprender las diferentes unidades con las que consta la asignatura de forma permanente y continua.
- ✓ El cd- interactivo es un documento que es un éxito para la catedra de desarrollo de sistemas y para los alumnos porque ellos se basaran en él para aprender e interpretar los pasos a seguir para desarrollar la asignatura, beneficiando así a docentes como también a estudiantes.

BIBLIOGRAFIA

González Alvarado, C. (1996). *Sistemas de Bases de Datos. (Primera Edición)*
Costa Rica: Tecnológica de Costa Rica.

A. O' Brien, J.; M. Marakas, G. (2006). *Sistemas de Información Gerencial (Séptima Edición)* México D. F.: Mc Graw Hill.

Pressman, R. S. (2005). *Ingeniería del Software, Un enfoque práctico (séptima edición)* México D. F.: Mc Graw Hill.

Lobel, L., Brust, A. J. (2008). *Programing Microsoft SQL Server 2008 (Primera Edición)* New York: Twentysix.

Ben-Gan, I. (2008). *Microsoft SQL Server 2008 T-SQL Fundamentals. (Primera Edicion)* Londres: Solid Quality.

MacDonald, M., Fremon A. and Szpuszta M. (2010). *Pro ASP.NET 4 in C# 2010 (Primera Edición)* USA: Apress.

Ing. Callejas E. A. (Ciclo 02-2011). Cuaderno de clases. *Desarrollo de Sistemas Informáticos Web II*. El Salvador: Universidad Tecnológica de El Salvador.

CCIA S. (Octubre 2008). *"Un Servicio Web"*. Recuperado de <http://ccia.ei.uvigo.es/docencia/SCS/0910/transparencias/Tema4.pdf>

Alegsa (2012). *Diccionario de Informática*. Recuperado de <http://www.alegsa.com.ar/Dic/acid.php>

Alegsa (2012). *Diccionario de Informática*. Recuperado de <http://www.alegsa.com.ar/Dic/sistema%20transaccional.php>

Dradi7, (junio-2008). *¿Qué es Linq?* [Registro web] Recuperado de <http://www.forosdelweb.com/f29/que-linq-711430/>

Jiménez A. (10 septiembre 2007) *Linq. ¿Qué es y como usarlo?* Recuperado de <http://alexjimenez.wordpress.com/2007/09/10/linq-definicion-como-usarlo/>

Microsoft (12/14/2011). *Introduction to Membership*. [Registro web] Recuperado de <http://msdn.microsoft.com/en-us/library/yh26yfzy.aspx>

MNLAntony, (Mar 27, 2012). *Conectar a BD de SQL Server 2008 con Visual Studio 2010 en ASP.NET*. [Registro web] Recuperado de <http://www.youtube.com/watch?v=4Pj7JJ2vWaw&feature=related>

Sabinadicto, (May 8, 2011). *Insertar, Borrar y Actualizar datos utilizando ADO.Net 3.5*. [Registro web] Recuperado de <http://www.youtube.com/watch?v=vSNWAXNF02U&feature=fvwrel>

ANEXO 1: CRONOGRAMA DE ACTIVIDADES

FASES	FEBRERO				MARZO				ABRIL				MAYO			
	Se- ma- na 1	Se- ma- na 2	Se- ma- na 3	Se- ma- na 4	Se- ma- na 1	Se- ma- na 2	Se- ma- na 3	Se- ma- na 4	Se- ma- na 1	Se- ma- na 2	Se- ma- na 3	Se- ma- na 4	Se- ma- na 1	Se- ma- na 2	Se- ma- na 3	Se- ma- na 4
CAPÍTULO I: SITUACION ACTUAL																
- Situación Problemática.																
- Justificación.																
- Objetivos.																
- Alcances																
- Estudio de la Factibilidad.																
CAPÍTULO II: DOCUMENTACIÓN TÉCNICA																
- Recopilación de Requisitos.																
- Recopilación de requisitos Técnicos.																
- Instalación y configuración de herramientas.																
CAPÍTULO III: DISEÑO DE LA SOLUCIÓN.																
- Diseño del Material Didáctico.																
- Diseño de Ejemplos Prácticos.																
- Diseño de ejercicios Propuestos.																

ANEXO 2: PRESUPUESTO

El recurso económico para la elaboración del proyecto proviene de fondos personales.

FINANCIERO HUMANO

No.	DETALLE	CANTIDAD	TOTAL
1	Cuotas de seminario de graduación	10	571.4
2	Transporte	80	32
3	Alimentación	20	40
	Sub total		643.4

FINANCIERO TECNOLÓGICO

No.	DETALLE	CANTIDAD	TOTAL
1	Impresiones	5	15
2	Fotocopias	10	15
3	Anillados	5	7.50
	Sub total		37.50

FINANCIAMIENTO MATERIAL

No.	DETALLE	CANTIDAD	TOTAL
1	Resma de papel bond	1	3.50
2	Tinta	2	20
	Sub total		23.50

COSTO DEL PROYECTO DE GRADUACIÓN

No.	DETALLE	TOTAL
1	Financiero humano	643.4
2	Financiero tecnológico	37.50
3	Financiero material	23.50
4	Imprevistos 20%	84.6
	Total	789.00

ANEXO 3: ENTREVISTA #1 (Dirigida a Estudiantes de Desarrollo de Sistemas)

Objetivo: Investigar la Situación Problemática que poseen los estudiantes de la asignatura de Desarrollo de Sistemas Informáticos Web II.

1. ¿Ya curso las asignaturas de Base de Datos II?

2. ¿Curso la asignatura de Programación II?

3. ¿Consulta usted la biblioteca central de la Universidad?

4. ¿Encuentra en la biblioteca información concisa?

5. ¿Consulta usted bibliotecas virtuales?

6. ¿Posee internet en su casa?

7. ¿Encuentra en internet el tema que busca?

8. ¿Cree usted como estudiante de técnico en ingeniería de software que se debería crear un material que abarque todas las temáticas de las asignaturas?

ANEXO 4: ENTREVISTA #2 (Dirigida a Estudiantes de Desarrollo de Sistemas)

Objetivo: Investigar los conocimientos que los estudiantes posee de las herramientas de MS SQL Server 2008 y Visual Studio 2010.

1. ¿Maneja MS SQL Server 2008 a nivel de procedimientos almacenados?

2. ¿Maneja MS SQL Server 2008 a nivel de disparadores o triggers?

3. ¿Maneja MS SQL Server 2008 a nivel de cursores?

4. ¿Maneja la herramienta Linq?

5. ¿Tiene conocimiento acerca de WebServices?

6. ¿Conoce usted las herramientas de Ajax?

7. ¿Ha utilizado usted controles de Ajax?

8. ¿Ha integrado una base de datos de SQL Server con un proyecto web de Microsoft Visual Studio?

ANEXO 5: ENTREVISTA #3 (Dirigida a Docentes de Desarrollo de Sistemas)

Objetivo: Realizar una investigación a los docentes sobre los beneficios que se obtendrían a elaborar un cd interactivo que abarque todas las temáticas de la asignatura Desarrollo de Sistemas Informáticos Web II.

1. ¿Considera usted como docente que la elaboración de un cd interactivo favorecería el aprendizaje de estudiante?

2. ¿Cuál unidad considera usted que debería de reforzarse en el cd-interactivo para que no quede ninguna duda en los estudiantes?

3. ¿Cómo cree usted conveniente que se le proporcione el cd interactivo al estudiante ya sea con algún costo o gratuitamente?

4. ¿cree que es conveniente que el cd interactivo este elaborado a base de videos interactivos para que los estudiantes se familiaricen pasos a pasos el desarrollo de las unidades?

5. ¿Considera necesario que se elaboren guías de ejercicio en el cd interactivo para que los estudiantes practiquen los temas desarrollados en la asignatura?
