

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA, LEÓN
UNAN-LEÓN

FACULTAD DE CIENCIAS Y TECNOLOGÍA

Departamento de Computación



Trabajo de Tesis de Maestría Presentado Para Optar a Título de:

Master en Tecnologías de la Informática Empresarial

Sistema informático para la gestión de los procesos de compras en el área de Adquisiciones de la UNAN-León.

Autor:

Ing. Bayardo José Cortez Castillo
Ing. Juan Carlos Leytón Briones

Tutor:

MSc. Rina Arauz

León, Nicaragua 2015

Dedicatoria y Agradecimiento

Primeramente a Dios por darnos la fuerza y sabiduría para seguir adelante en los momentos difíciles guiarnos por el camino correcto.

A Nuestros padres que nos enseñan a luchar por nuestros sueños y que siempre nos apoyan con sus consejos y bendiciones.

A nuestros hijos que son nuestra felicidad, nuestra fuente de inspiración y querer ser para ellos un ejemplo a seguir.

A Nuestra tutora MSc. Rhina Arauz por su dedicación y empeño en orientarnos para que este trabajo culminara con éxito.

Tabla de contenido

I. Resumen	6
II. Introducción.....	7
2.1 Planteamiento del problema	7
2.2 Antecedentes	8
2.3 Justificación.....	9
2.4 Objetivos	11
2.4.1 Objetivo General.....	11
2.4.2 Objetivos Específicos	11
2.5 Resultados esperados	12
2.6 Productos Obtenidos	12
2.6.1 Los archivos .aspx	12
2.6.2 Los archivos .aspx.cs	12
2.6.3 Los archivos .cs que definen clases no vinculadas a ninguna página	13
III. Tecnologías Empleadas.....	13
3.1 Visual Studio	13
3.2 ASP.NET.....	13
3.2.1 Marco de trabajo de páginas y controles	14
3.2.2 Compilador de ASP.NET	15
3.2.3 Infraestructura de seguridad	15
3.2.4 Funciones de administración de estado	16
3.2.5 Configuración de ASP.NET	16
3.2.6 Supervisión de estado y características de rendimiento	16
3.2.7 Capacidad de depuración	17
3.2.8 Marco de trabajo de servicios Web XML.....	17
3.2.9 Entorno de host extensible y administración del ciclo de vida de las aplicaciones	17
3.2.10 Entorno de diseñador extensible.....	18
3.3 Internet Information Server.....	18
3.4 Microsoft Sql Server:.....	18
3.5 JavaScript	19
3.6 HTML.....	19
3.7 CSS.....	19
IV. Diseño Metodológico	20
4.1 Análisis de requisitos	20

4.2 Diseño del Sistema.....	20
4.3 Diseño del programa.....	20
4.4 Codificación.....	20
4.5 Pruebas	20
4.6 Verificación	20
V. Análisis de requisitos	22
5.1 Alcance.....	22
5.2 Modelo de Casos de Uso	22
5.3 Descripción de actores.....	24
5.3.1 Actor Solicitante.....	24
5.3.2 Actor Presupuesto	24
5.3.3 Actor Analista.....	25
5.3.4 Actor Administrador	25
5.4 Descripción de casos de uso	26
5.4.1 Ingresar solicitudes	26
5.4.2 Ingresar usuarios.....	26
5.4.3 Ingresar Facultad, Dependencia, Fondo, Proveedor	26
5.4.4 Administrar todos los módulos.....	26
5.4.5 Aprobar Solicitud	26
5.4.6 Cambiar clave.....	27
5.4.7 Generar Orden de compra.....	27
5.4.8 Generar reportes e informes	27
5.5 Recursos hardware y software	27
5.5.1 Materiales Hardware	27
5.5.2 Materiales Software.....	28
VI. Diseño de la aplicación	28
6.1 Especificaciones de usabilidad.....	28
6.2 Diseño de las Páginas.....	28
6.2.1 Opción Solicitud	29
6.2.2 Opción Presupuesto.....	29
6.2.3 Opción Usuario	29
6.2.4 Opción Proveedor	30
6.2.5 Opción Orden de Compra	30
6.2.6 Opciones Facultad, Dependencias, Fondos	30
6.2.7 Opción Cambiar Contraseña	30

6.3 Estructura de la Aplicación:.....	30
6.3.1 Página Index.aspx	30
6.3.2 Página Principal.aspx	30
6.3.3 Diseño de la base de datos	30
VII. Implementación de la aplicación.....	36
7.1 Estructura de la aplicación.....	36
7.2 Codificación de la Conexión a la base de datos Compras.....	37
7.3 Herramientas Externas	38
7.3.1 Uso de JavaScript	38
7.3.2 Uso de Css	39
7.4 Seguridad en la Aplicación.....	39
7.4.1 Autenticación de usuario al ingresar a la aplicación	39
7.4.2 Validación de las variables de sesión.....	39
7.4.2 Encriptación de clave	39
VIII. Publicación.....	40
8.1 Publicar aplicación web en Internet Information Server.....	40
IX Conclusiones y trabajos futuros.....	42
9.1 Conclusiones	42
9.2 Líneas de trabajos futuros	42
X. Manual de la aplicación	44
10.1 Ingreso al sistema	45
10.2 Ingresar Facultad y Dependencia.....	46
10.4 Ingresar Usuario.....	48
10.5 Cambio de clave de Usuario	50
10.3 Ingresar Departamentos y Municipios:	51
10.6 Administradores del Sistema:.....	53
XI. Bibliografía.....	54

I. Resumen

En la UNAN-León, la Dirección de Adquisiciones es el área encargada de realizar los procedimientos de compras de bienes, servicios u obras de todas las Dependencias, Facultades y Centros Regionales, los usuarios finales administradores, docentes, directores, decanos se quejan que existe una problemática en cuanto al tiempo que tardan los procesos de compra, en cuanto a la calidad de materiales e insumos adquiridos y el alto costo de los mismos.

Considerando la problemática expuesta este proyecto de tesis plantea el desarrollo de una alternativa que ayude a mejorar los rendimientos en los procesos de compras y disminuir los plazos de entrega de los materiales solicitados con apoyo de una aplicación web que permita gestionar, controlar y dar seguimientos a todas las actividades que se realizan en el área de Adquisiciones. La aplicación estará enfocada en dar seguimiento a las solicitudes de compra que realizan todos los usuarios con el fin de evitar retrasos y entregar los productos o materiales en tiempo razonable.

La infraestructura de la red, el equipamiento de equipos en las oficinas así como el acceso a internet de muy buen ancho de banda, hace de este proyecto una opción interesante para valorar su aceptación, dicha aplicación web se estaría implementando en Lenguaje de programación Microsoft Visual C# utilizando el Framework ASP.NET y Microsoft Sql Server 2012 como Gestor de Bases de Datos.

II. Introducción

2.1 Planteamiento del problema

La Dirección de Adquisiciones de la UNAN-León, es la unidad encargada de realizar los procedimientos de compra establecidos en la Ley 737 "Ley de contrataciones administrativas del sector público", las solicitudes de compra son realizadas por las diferentes Facultades, Dependencias y CUR regionales que conforman la Institución.

Desde el mes de Marzo el año 2011 que entro en vigencia la ley 737 se realizó la centralización de las compras y desde esa fecha es responsabilidad de la Unidad de Adquisiciones cumplir con los procedimientos establecidos por esta Ley.

Las actividades de la Unidad de Adquisiciones es recepcionar y atender las solicitudes de compra, aplicando el procedimiento establecido en la ley, el que incluye las siguientes actividades: realizar invitación a proveedores, recepción ofertas, evaluación de ofertas, realización de actas de evaluación, realización de orden de compra y envió de dicha orden a los proveedores asignados. Los problemas más relevantes que los usuarios solicitantes señalan de la Unidad de Adquisiciones:

El tiempo de entrega de los productos solicitados, el producto entregado no corresponde con lo esperado y los costos Elevados.

Ante esta problemática se propone el desarrollo de una aplicación web¹ que ayude a disminuir los inconvenientes planteados, este desarrollo abarcaría todas las etapas de los procesos de compra (realización de la solicitud, aprobación por la oficina de presupuesto, recepción de documentos en dirección de adquisiciones, generación de orden de compra, generación de Informes, consultas entre otras funciones), está aplicación sería desarrollada utilizando el lenguaje de

¹ Aplicación Web: es aquella en que los usuarios acceden a ellas en un servidor Web a través de Internet o de una intranet (Pereira 2006, GestioPolis, <http://www.gestiopolis.com/>)

programación Microsoft Visual C# 2013 utilizando el Framework² ASP.NET y Gestor de Bases de Datos Microsoft SQL Server 2012 .

2.2 Antecedentes

Desde el año 2011 la Unidad de Adquisiciones cuenta con un personal de 10 trabajadores donde se desempeñan cargos de director, responsables de contratos, responsables de compras, analistas y contador, desde esta fecha esta Dependencia no contaba con ninguna herramienta informática para la gestión de los procesos de compra, toda solicitud realizada por administradores de facultades o directores de dependencias se realizaba en Word u otro editor de texto y era enviada a la Unidad de Adquisiciones, la realización de la Orden de compra se realizaba en Máquina de escribir.

En el año 2012 a petición de la dirección de adquisiciones, la División de Informática desarrolla una aplicación web la cual se ejecuta únicamente en la red interna de la UNAN-León con el objetivo de automatizar los procedimientos y tareas de los trámites de compra, esta aplicación web fue implementada utilizando las herramientas de software libre³ PHP como lenguaje de programación y Mysql como Gestor de Base de datos.

Actualmente la aplicación sigue en uso y entre las principales funcionalidades está Ingresar Solicitudes de compra, generar orden de compra, reportes e informes entre otros, pero dicha aplicación presenta algunas debilidades ya que no cubre todos los aspectos o etapas de los procedimientos de compra.

² Framework: es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado (Alegsa 2010, <http://www.alegsa.com.ar/Dic/framework>).

³ Aquel que respeta la libertad de los usuarios sobre el producto, es decir que puede ser usado, copiado, estudiado, modificado y redistribuido libremente (Stallman, 2011)

2.3 Justificación

Ante la problemática expuesta se propone una alternativa que ayude a mejorar el desempeño y rendimiento de las funciones del área de Adquisiciones, esta propuesta es desarrollo de una aplicación web destinada a la Gestión, Control y Seguimiento de los procesos de compra en la Dirección de Adquisiciones de la UNAN-León, la que automatizaría algunas etapas de los procesos, brindaría alertas en plazos de etapas vencidas, reporte de procesos que deberían haber culminado según los plazos establecidos entre otras funcionalidades.

Esta aplicación proporcionaría numerosos beneficios en las funciones de la Dirección de Adquisiciones mejorando el rendimiento, disminuyendo los plazos en la ejecución de los procesos, recuperando la satisfacción de los usuarios solicitantes y apoyando al buen funcionamiento de la Institución Universitaria.

Considerando el avance de la tecnología actual así como la variedad de aplicaciones para el desarrollo de sistemas tanto de escritorio como web, el equipamiento en las oficinas administrativas de la institución, analizando la infraestructura de la red, contamos con conexión entre recintos con fibra óptica y un ancho de banda de 200 megabytes, esto nos proporciona los usuarios una conexión estable y de muy buena velocidad en todos recintos universitarios incluyendo los CUR (Centros Universitarios Regionales), se presenta el proyecto como una alternativa que ayude a mejorar las el rendimiento de las funciones del personal de adquisiciones.

Está aplicación sería desarrollada utilizando el lenguaje de programación Visual C# .NET con el Framework ASP.NET, utilizando el gestor de base de datos Microsoft SQL Server 2012, se selecciona Visual C# por ser un lenguaje moderno, orientado a objetos simple, intuitivo, muy eficiente y con seguridad en el tratamiento de tipos para generar una gran variedad de aplicaciones⁴.

Existen en Internet algunas herramientas comerciales de pago destinadas a gestionar las compras, la adquisición de una de estas aplicaciones no serían de mucho beneficios ya que no permitiría cubrir todas las etapas que se realizan en el

⁴ Guerrero, M. F. (s.f.). *Microsoft*. Recuperado el 5 de Marzo de 2015, de <https://msdn.microsoft.com/es-es/library/bb972208.aspx>

área de Adquisiciones además del costo que habría que invertir. La Metodología a utilizar es modelo en Cascada⁵ debido a que permite el desarrollo de las actividades de manera secuencial o lineal logrando una independencia en las etapas de desarrollo.

⁵Guerrero, M. F. (s.f.). *Microsoft*. Recuperado el 5 de Marzo de 2015, de <https://msdn.microsoft.com/es-es/library/bb972208.aspx>

2.4 Objetivos

2.4.1 Objetivo General

Desarrollar una aplicación web en Visual C# utilizando el Framework ASP.NET para la gestión de los procesos de compras de la Dirección de Adquisiciones de la UNAN-León, en el período de Marzo a Septiembre 2015.

2.4.2 Objetivos Específicos

- Realizar análisis del problema para comprender las diferentes fases y ciclos de los procesos de compras, realizando entrevista con el personal de las áreas involucradas.
- Implementar el diseño de la aplicación para las etapas de desarrollo y efectuar cada uno de ellas en orden secuencial.
- Realizar la codificación de la aplicación utilizando el lenguaje de programación seleccionado para el desarrollo de la aplicación.
- Ejecutar diagnóstico de la aplicación con pruebas de funcionamiento para que se cumpla con todas las necesidades requeridas por los usuarios finales.

2.5 Resultados esperados

- Información necesaria para la implementación de cada uno de los módulos que conforman la aplicación.
- Tener el diseño de mi aplicación web definida en cada una de los módulos de desarrollo.
- Aplicación web Desarrollada.
- Aplicación web en prueba cumpliendo con los requerimientos de los usuarios finales.

2.6 Productos Obtenidos

Como resultado del trabajo se han obtenido los siguientes productos: por un lado el código de la aplicación Web implementada y por el otro toda la documentación generada en las fases de análisis y diseño.

El código se divide en los siguientes archivos:

2.6.1 Los archivos .aspx

Donde está el código que define los Web Forms o interfaz gráfica de cada página. Los ficheros son: ccontrasesña.aspx, CerrarSesion.aspx, Dependencia.aspx, Index.aspx, Principal.aspx, Prueba.aspx, Solicitud.aspx, Usuario.aspx, OrdenCompra.aspx.

2.6.2 Los archivos .aspx.cs

Donde está el código que implementa las funcionalidades de cada página, los ficheros son: ccontrasesña.aspx.cs, CerrarSesion.aspx.cs, Dependencia.aspx.cs, Index.aspx.cs, Principal.aspx.cs, Prueba.aspx.cs, Solicitud.aspx.cs, Usuario.aspx.cs, OrdenCompra.aspx.cs

2.6.3 Los archivos .cs que definen clases no vinculadas a ninguna página

Los ficheros son: Conexion.cs clase encargada de realizar las conexiones a nuestra base de datos y el archivo Web.config en el que se guarda la configuración general de la aplicación web.

III. Tecnologías Empleadas

A Continuación de describen las tecnologías más relevantes utilizadas para el desarrollo de este proyecto:

3.1 Visual Studio

Visual Studio es todo un conjunto de herramientas de desarrollo para la generación de aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles. Visual Basic, Visual C# y Visual C++ utilizan todos el mismo entorno de desarrollo integrado (IDE), que habilita el uso compartido de herramientas y hace más sencilla la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes utilizan las funciones de .NET Framework, las cuales ofrecen acceso a tecnologías clave para simplificar el desarrollo de aplicaciones web ASP y Servicios Web XML.

3.2 ASP.NET

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, con el código mínimo. ASP.NET forma parte de .NET Framework y al codificar las aplicaciones ASP.NET tiene acceso a las clases en .NET Framework. El código de las aplicaciones puede escribirse en cualquier lenguaje compatible con el Common Language Runtime⁶ (CLR), entre ellos Microsoft Visual Basic, C#, JScript .NET y J#. Estos lenguajes permiten desarrollar

⁶ **CLR (Common Language Runtime)**

Maneja el código en tiempo de ejecución y proporciona todos los servicios básicos para su correcto funcionamiento.

Es un entorno administrativo que provee aspectos como servicios comunes, garbage collection y seguridad. Es el motor de ejecución. El CLR ofrece servicios como:

1. Administración de la memoria
2. Seguridad del código
3. Versionamiento

aplicaciones ASP.NET que se benefician del Common Language Runtime, seguridad de tipos, herencia, etc.

ASP.NET incluye:

- Marco de trabajo de página y controles
- Compilador de ASP.NET
- Infraestructura de seguridad
- Funciones de administración de estado
- Configuración de la aplicación
- Supervisión de estado y características de rendimiento
- Capacidad de depuración
- Marco de trabajo de servicios Web XML
- Entorno de host extensible y administración del ciclo de vida de las aplicaciones
- Entorno de diseñador extensible

3.2.1 Marco de trabajo de páginas y controles

El marco de trabajo de páginas y controles ASP.NET es un marco de trabajo de programación que se ejecuta en un servidor Web para generar y representar de forma dinámica páginas Web ASP.NET. Las páginas Web ASP.NET se pueden solicitar a cualquier explorador o dispositivo del cliente y ASP.NET representa el marcado (como HTML) al explorador que realizó la solicitud. Como norma, puede utilizar la misma página para varios exploradores, porque ASP.NET representa el marcado adecuado para el explorador que realiza la solicitud. Sin embargo, puede diseñar una página Web ASP.NET para ejecutarse en un explorador determinado, como Microsoft Internet Explorer 6, y aprovechar así todas las características de ese explorador. ASP.NET es compatible con los controles móviles de los dispositivos preparados para trabajar en Web como teléfonos celulares, PC portátiles y asistentes digitales personales (PDA).

Las páginas Web ASP.NET están completamente orientadas a objetos. Se puede trabajar con elementos HTML que usen propiedades, métodos y eventos. El marco de trabajo de páginas ASP.NET quita los detalles de implementación relacionados con la separación de cliente y servidor inherente a las aplicaciones Web presentando un modelo unificado que responde a los eventos de los clientes en el código que se ejecuta en el servidor. El marco de trabajo también mantiene automáticamente el estado de la página y de los controles que contenga durante el ciclo vital de procesamiento de la página.

El marco de trabajo de páginas y controles ASP.NET también permite encapsular la funcionalidad común de la interfaz de usuario en controles fáciles de usar y

reutilizables. Los controles se escriben una vez, se pueden utilizar en varias páginas y se integran en la página Web ASP.NET en la que se colocan durante la representación.

El marco de trabajo de páginas y controles ASP.NET también proporciona funciones para controlar la apariencia y el funcionamiento general de los sitios Web a través de temas y máscaras. Se pueden definir temas y máscaras y, a continuación, aplicarlos en las páginas o controles

Además de los temas, es posible definir páginas principales que se crean para conseguir un diseño coherente en las páginas de la aplicación. Una página principal única define el diseño y el comportamiento estándar deseados para todas las páginas (o un grupo de páginas) de la aplicación. A continuación, se pueden crear páginas de contenido individuales con el contenido específico de la página que se desee mostrar. Cuando los usuarios solicitan las páginas de contenido, las combinan con la página principal con el fin de generar un resultado que combine el diseño de la página principal con el de la página de contenido.

3.2.2 Compilador de ASP.NET

Compila todo el código de ASP.NET, lo que permite el establecimiento inflexible de tipos, las optimizaciones de rendimiento y el enlace en tiempo de compilación, entre otras ventajas. Una vez que se ha compilado el código, el Common Language Runtime compila una vez más código de ASP.NET en código nativo, lo que permite un mayor rendimiento.

3.2.3 Infraestructura de seguridad

Además de las características de seguridad de .NET, ASP.NET proporciona una infraestructura de seguridad avanzada para autenticar y autorizar el acceso de los usuarios y realizar otras tareas relacionadas con la seguridad. Puede autenticar usuarios con la autenticación de Windows suministrada por IIS o puede administrar la autenticación con su propia base de datos de usuario utilizando la autenticación mediante formularios ASP.NET y la suscripción ASP.NET. Además, puede administrar la autorización a las capacidades e información de su aplicación Web mediante los grupos de Windows o su propia base de datos de funciones personalizada utilizando las funciones de ASP.NET. Resulta fácil quitar, agregar o reemplazar estos esquemas dependiendo de las necesidades de la aplicación. Para obtener más información, vea los temas siguientes:

- Proteger sitios web ASP.NET
- Administrar usuarios mediante suscripciones
- Administrar autorizaciones con funciones
- Proveedor de autenticación mediante formularios

ASP.NET siempre se ejecuta con una identidad particular de Windows de modo que puede asegurar su aplicación utilizando las capacidades de Windows como, por ejemplo, las listas de control de acceso (ACL) de NTFS, permisos de la base de datos, etc. Para obtener más información sobre la identidad de ASP.NET, vea Configurar la identidad de procesos en ASP.NET y Suplantación de ASP.NET.

3.2.4 Funciones de administración de estado

ASP.NET proporciona funcionalidad de administración de estado intrínseca que permite almacenar información entre las solicitudes de página, como la información de clientes o el contenido del carro de la compra. Puede guardar y administrar información específica de la aplicación, específica de la sesión, específica de la página, específica del usuario y definida por el desarrollador. Esta información puede ser independiente de cualquier control de la página,

ASP.NET ofrece funciones de estado distribuidas, lo que le permite administrar información de estado en múltiples instancias de la misma aplicación en un equipo o en varios.

3.2.5 Configuración de ASP.NET

Las aplicaciones ASP.NET utilizan un sistema de configuración que le permite definir valores de configuración para su servidor Web, para un sitio Web o para aplicaciones individuales. Puede crear valores de configuración cuando se implementan las aplicaciones ASP.NET y puede agregar o revisar los valores de configuración en cualquier momento con un impacto mínimo en aplicaciones y servidores Web de operaciones. Los valores de configuración de ASP.NET se almacenan en archivos basados en la tecnología XML. Dado que estos archivos XML son archivos de texto ASCII, es fácil realizar cambios de configuración a sus aplicaciones Web. Puede extender el esquema de configuración para satisfacer sus requisitos.

3.2.6 Supervisión de estado y características de rendimiento

ASP.NET incluye características que le permiten supervisar el estado y el rendimiento de su aplicación ASP.NET. La supervisión del estado de ASP.NET permite proporcionar información sobre eventos clave que proporcionan información sobre el estado de una aplicación y sobre las condiciones de error. Estos eventos muestran una combinación de diagnósticos y características de supervisión, a la vez que proporcionan un elevado grado de flexibilidad en lo que respecta a lo que se registra y cómo.

ASP.NET admite dos grupos de contadores de rendimiento a los que pueden obtener acceso las aplicaciones:

- El grupo de contadores de rendimiento del sistema ASP.NET
- El grupo de contadores de rendimiento de la aplicación ASP.NET

3.2.7 Capacidad de depuración

ASP.NET aprovecha la infraestructura de depuración en tiempo de ejecución para permitir la depuración entre lenguajes y equipos. Se pueden depurar tanto objetos administrados como no administrados, así como todos los lenguajes compatibles con el Common Language Runtime y los lenguajes de script.

Además, el marco de trabajo de páginas ASP.NET proporciona un modo de seguimiento que permite insertar mensajes de instrumentalización en las páginas Web ASP.NET.

3.2.8 Marco de trabajo de servicios Web XML

ASP.NET es compatible con los servicios Web XML. Un servicio Web XML es un componente que incluye funcionalidad de empresa que permite a las aplicaciones intercambiar información entre firewalls utilizando estándares como los servicios de mensajería HTTP y XML. Los servicios Web XML no están relacionados con ninguna tecnología de componentes ni con ninguna convención de llamada a objetos en concreto. Como resultado, pueden obtener acceso a los servicios Web XML los programas escritos en cualquier lenguaje, que usen cualquier modelo de componentes y se ejecuten en cualquier sistema operativo.

3.2.9 Entorno de host extensible y administración del ciclo de vida de las aplicaciones

ASP.NET incluye un entorno de host extensible que controla el ciclo de vida de una aplicación desde el momento en que un usuario cualquiera tiene acceso a un recurso (como una página) en la aplicación hasta el momento en que se cierra la aplicación. Aunque ASP.NET se basa en un servidor Web (IIS) como un host de la aplicación, ASP.NET proporciona gran parte de la propia funcionalidad de host. La arquitectura de ASP.NET permite responder a los eventos de aplicación y crear controladores y módulos HTTP personalizados.

3.2.10 Entorno de diseñador extensible

ASP.NET incluye la compatibilidad mejorada para crear diseñadores de controles de servidor Web para utilizarlos con una herramienta de diseño visual como Visual Studio. Los diseñadores permiten crear una interfaz de usuario en tiempo de diseño para un control; de este modo, los desarrolladores pueden configurar las propiedades y el contenido del control en una herramienta de diseño visual.

3.3 Internet Information Server

Son los servicios de software que admiten la creación, configuración y administración de sitios Web, además de otras funciones de Internet. Los servicios de Microsoft Internet Information Server incluyen el Protocolo de transferencia de noticias a través de la red (NNTP), el Protocolo de transferencia de archivos (FTP) y el Protocolo simple de transferencia de correo (SMTP).

3.4 Microsoft Sql Server:

SQL Server es un sistema de administración, gestión y análisis de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact -SQL), y es capaz de poner grandes cantidades de información a muchos usuarios simultáneamente y de manera muy rápida posee un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de

transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas.

Sql Server en sus últimas versiones es capaz de gestionar cualquier tipo de datos, en cualquier sitio y momento, almacena datos de documentos estructurados como imágenes, música y archivos directamente en la base de datos, se obtiene más rendimiento de los datos, poniendo a disposición servicios integrados como son consultas, búsquedas, sincronizaciones, informes y análisis.

3.5 JavaScript

JavaScript es un lenguaje ligero e interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C, conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js Apache CouchDB. Es un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa.

Todos los navegadores interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DOM.

3.6 HTML

Es un lenguaje que se utiliza en el desarrollo de páginas web. HTML son siglas de HyperText Markup Language (Lenguaje de Marcación de Hipertexto) es un lenguaje que se utiliza comúnmente para establecer la estructura y contenido de un sitio web, tanto de texto, objetos e imágenes. Los archivos desarrollados en HTML usan la extensión .html

El lenguaje de HTML funciona por medio de "etiquetas" que describen la apariencia o función del texto enmarcado. Este lenguaje puede llegar a incluir un script o código que tenga incidencia en el comportamiento del navegador web de elección.

3.7 CSS

Es un lenguaje utilizado en la presentación de documentos HTML. Un documento HTML viene siendo coloquialmente "una página web". Entonces podemos decir que el lenguaje CSS sirve para organizar la presentación y aspecto de una página web. Este lenguaje es principalmente utilizado por parte de los navegadores web de internet y por los programadores web informáticos para elegir multitud de opciones de presentación como colores, tipos y tamaños de letra, etc. Css es la herramienta que nos permite que el texto HTML se vea de una u otra manera.

IV. Diseño Metodológico

Existen muchas metodologías de desarrollo unas más recientes que otras su uso e implementación depende del desarrollador o programador, en este proyecto se utilizará una de las metodologías de programación tradicional como es el modelo en cascada también conocida como programación lineal o secuencial, en esta metodología cada una de las etapas de desarrollo de la aplicación se ejecuta de manera secuencial una después de culminar la anterior. Las etapas del modelo en cascada son:

4.1 Análisis de requisitos

En esta etapa se analizan las necesidades de los usuarios finales para determinar todo lo que se debe cubrir y tener claro todas las especificaciones de lo que se va hacer.

4.2 Diseño del Sistema

En esta etapa se divide el sistema en partes que puedan elaborarse por separado y describe que debe hacer cada una de sus partes así como la manera que se combinan unas con otras.

4.3 Diseño del programa

En esta etapa se desarrollan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario.

4.4 Codificación

Esta es la etapa donde se implementa todo el código fuente así como incluir todas las librerías necesarias por el lenguaje de programación seleccionado.

4.5 Pruebas

En esta etapa se realizan las pruebas correspondientes para comprobar que la aplicación funciona correctamente.

4.6 Verificación

Esta es la parte donde el usuario final ejecuta el sistema comprobando que no falla nada.

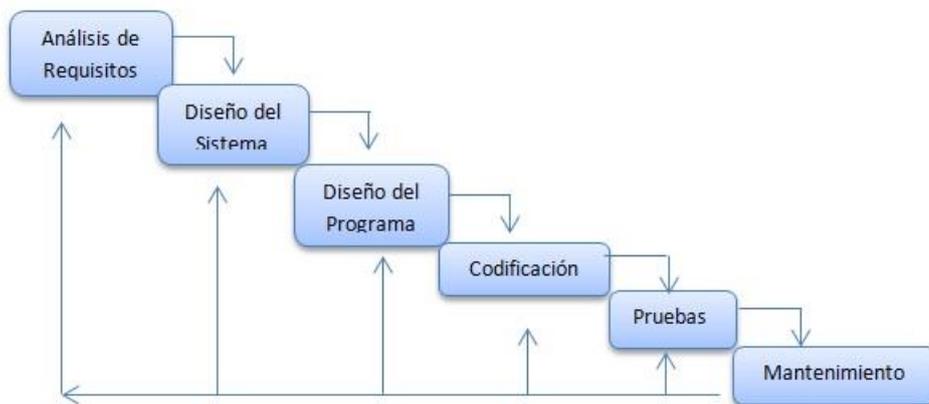


Fig. 1 fases del modelo en cascada

En la etapa de análisis de requisitos también conocida como recolección de datos se realizó entrevista con personal de presupuesto, administradores, coordinadores de proyectos y programas de igual forma con personal de Adquisiciones para conocer las actividades de cada una de estas áreas, determinar los requerimientos necesarios para el buen funcionamiento de la aplicación, en esta etapa se establecerán los perfiles que deberá tener la aplicación así como las funciones y restricciones de cada uno de ellos.

Se seleccionará un trabajador de cada área para que sea la encargada de brindar toda la información necesaria para definir las funciones y restricciones de ese perfil de usuario, en el caso del perfil usuario solicitante se tomarán dos trabajadores un administrador de facultad y un coordinador de proyecto para tomar ambas perspectivas y cumplir con ambos requerimientos, este personal será la muestra de este proyecto.

Con la información recopilada se comienza a realizar etapa de diseño de sistema para dividir la aplicación en partes que se puedan realizar por separadas pero que a la vez están relacionadas aquí se definen módulos, funciones, estructurar de la base de datos, tipos de reportes, consultas, mecanismo de seguridad, luego viene el diseño del programa donde crearemos los algoritmos de programación que nos servirán para realizar la etapa de codificación de toda la aplicación esto se realizará en el lenguaje de programación Microsoft Visual C# 2012 utilizando el Framework ASP.NET

La elaboración del proyecto se enmarca en toda la UNAN-León ya que esta aplicación será de uso para todas las Facultades, Dependencias, CUR Regionales, Programas y Proyecto aunque su mayor uso estará en la gestión y control de los procesos en el área de Adquisiciones.

La realización de este proyecto tendrá una duración de seis meses y su fecha de inicio es el 16 de Febrero 2015 y su fecha de finalización está prevista para el día 13 de Septiembre 2015.

V. Análisis de requisitos

El proceso del estudio de las necesidades de los usuarios para llegar a una definición de los requisitos del sistema, de hardware o de software, así como el proceso de estudio y refinamiento de dichos requisitos" (Estándar IEEE Std. 610 [IEEE 1990])

5.1 Alcance

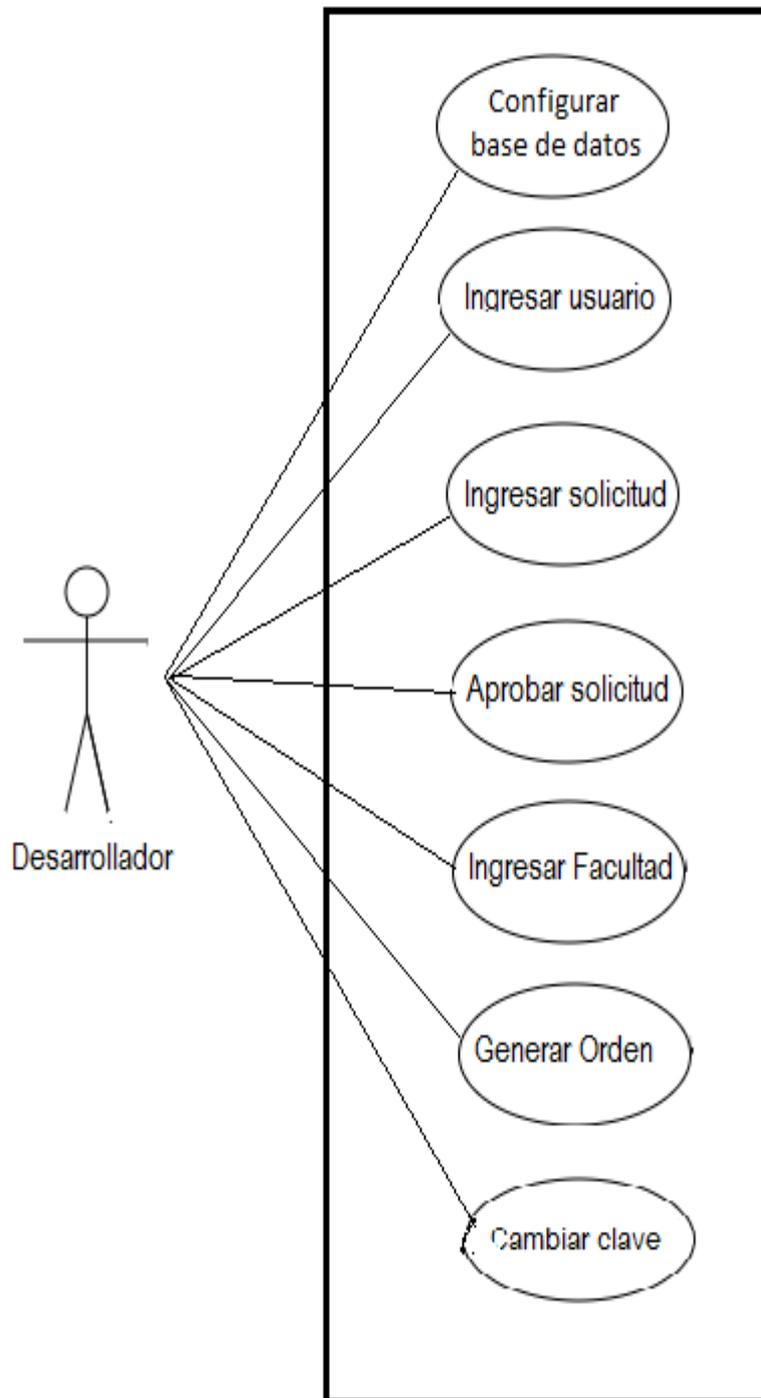
El nombre de esta aplicación web será: **Sistema de Gestión y Control de las Adquisiciones de la UNAN-León**. Y permitirá a los usuarios las siguientes funciones

- Autenticación de usuarios:
- Ingreso de solicitudes de compras.
- Ingresos de usuarios y proveedores.
- Aprobación de Solicitudes por presupuesto
- Ingresos de Facultades y Dependencias
- Generación de orden de compras.
- Generación de informes y reportes
- Cambio de clave de usuarios
- Cerrar Sesión

- Actualizar y borrar registros

5.2 Modelo de Casos de Uso

Es un modelo que describe los requerimientos funcionales de un sistema en términos de casos de uso, sirve para capturar todas necesidades del usuario obtenidas de una representación en forma gráfica.

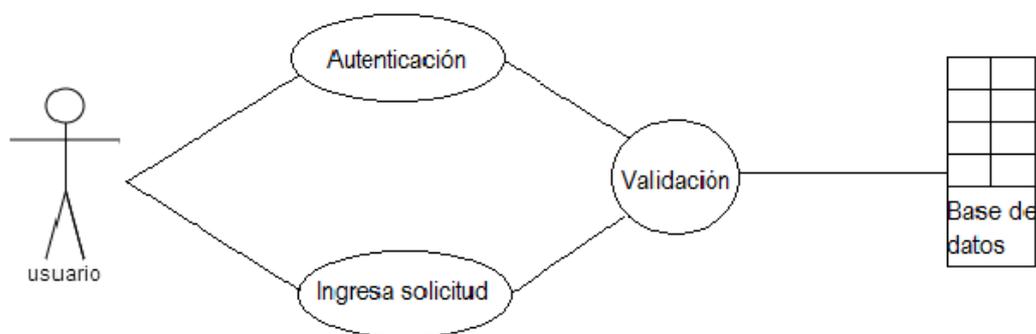


5.3 Descripción de actores

En la aplicación web se definen cuatro tipos de actores con el fin de controlar todos los procesos del sistema de gestión de adquisiciones.

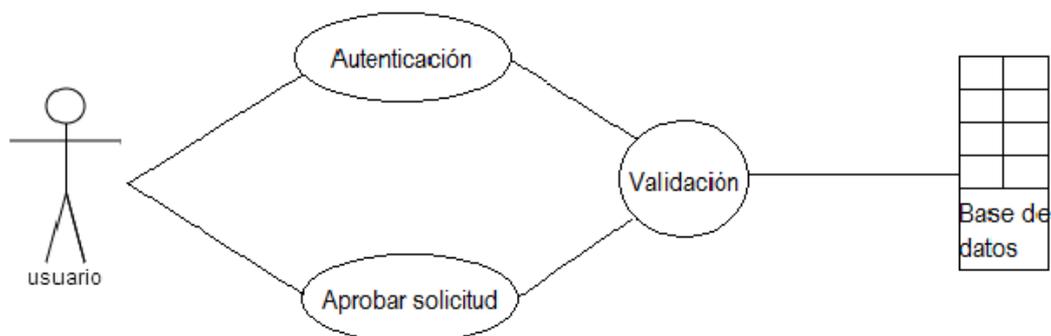
5.3.1 Actor Solicitante

Caso de uso: Ingresar al sistema, Ingresar productos, materiales, bienes o servicios, visualizar avance de la solicitud, generar reporte e informes.



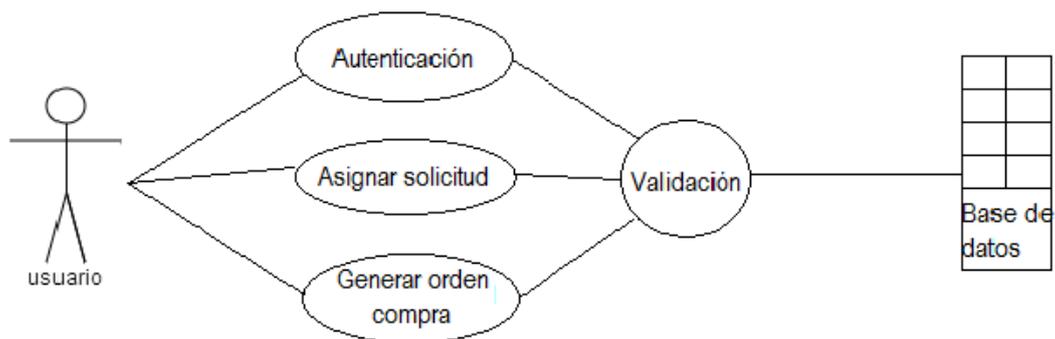
5.3.2 Actor Presupuesto

Caso de uso: Ingresar al sistema, visualizar las solicitudes, aprobar o rechazar solicitud, generar informes o reportes.



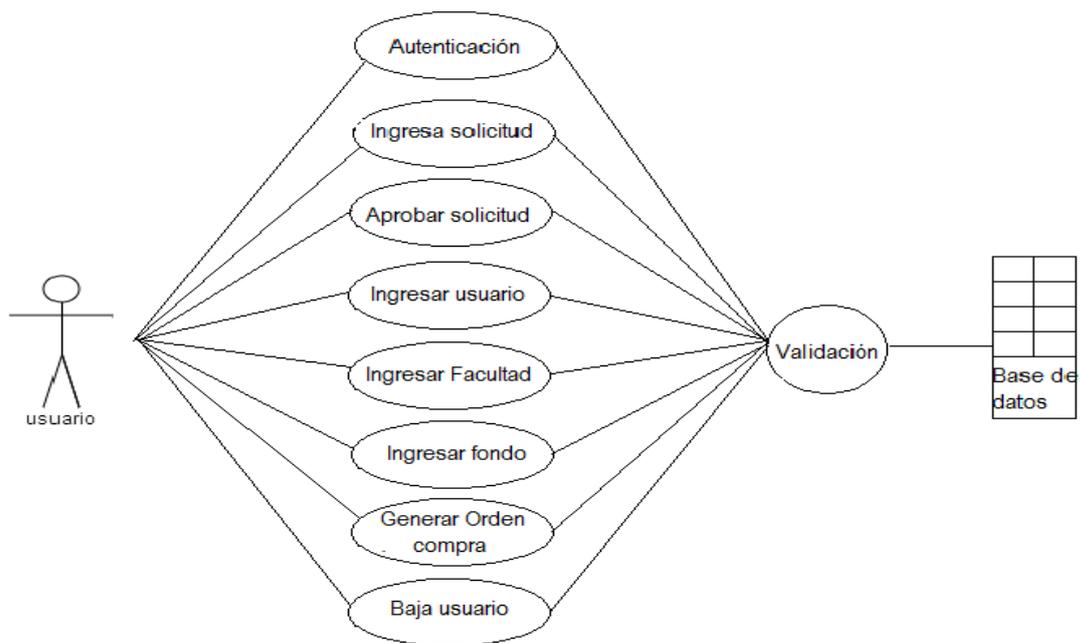
5.3.3 Actor Analista

Casos de uso: Ingresar al sistema, asignar solicitud, generar orden de compra, generar reporte e informes.



5.3.4 Actor Administrador

Casos de uso: Ingresar al sistema, crear usuarios, dependencias, fondos, proveedores, anular solicitud, orden de compra, baja de usuarios.



5.4 Descripción de casos de uso

5.4.1 Ingresar solicitudes

Permite ingresar los productos, bienes o servicios que se quieren adquirir, indicando cantidad, descripción, especificaciones, fondo, monto reservado.

- Presentación de la interfaz gráfica que permite ingresar los datos de la nueva solicitud.
- Validación correspondiente a la información ingresada en la aplicación
- Agregar la solicitud en la base de datos.

5.4.2 Ingresar usuarios

Permite ingresar nuevos usuarios a la aplicación

- Presentación de la interfaz gráfica que permite ingresar usuario
- Validación correspondiente a la información ingresada en la aplicación
- Agregar el usuario a la bases de datos

5.4.3 Ingresar Facultad, Dependencia, Fondo, Proveedor

Permite agregar a la bases de datos registros de tablas complementarias de la aplicación

- Presentación de la interfaz gráfica que permite Facultad, Dependencia, Fondo, Proveedor
- Validación correspondiente a la información ingresada en la aplicación
- Agregar el Facultad, Dependencia, Fondo, Proveedor a la bases de datos

5.4.4 Administrar todos los módulos

Permite realizar acciones de editar, actualizar y eliminar sobre todos los módulos de la aplicación

- Presentación de la interfaz gráfica que permite editar o eliminar.
- Obtener de la base de datos la información necesaria para la acción correspondiente
- Actualizar la información de forma correcta en la base de datos

5.4.5 Aprobar Solicitud

Permite autorizar una solicitud de compra, previa verificación del presupuesto disponible.

- Presentación de la interfaz gráfica para asignar solicitudes nuevas ingresadas
- Obtener de la bases de datos la información necesaria para aprobar la solicitud de compra
- Actualizar la información de forma correcta en la base de datos

5.4.6 Cambiar clave

Permite al usuario realizar un cambio de contraseña ingresando su contraseña actual e ingresando la nueva clave.

- Presentación de la interfaz gráfica que permite cambiar la clave
- Validación correspondiente a la información ingresada en la aplicación y comparación con datos almacenados
- Actualizar la información de forma correcta en la base de datos

5.4.7 Generar Orden de compra

Permite genera de la solicitud de compra su correspondiente orden de compra

- Presentación de la interfaz gráfica que permite generar orden de compra
- Extraer de la bases de datos la información necesaria para generar orden de compra
- Agregar la orden de compra de forma correcta en la base de datos

5.4.8 Generar reportes e informes

Permite visualizar los diferentes tipos de reportes e informes de la aplicación:

- Presentación de la interfaz gráfica que permite visualizar la información
- Extraer de la bases de datos la información necesaria para generar orden de compra.

5.5 Recursos hardware y software

5.5.1 Materiales Hardware

En el Desarrollo de la aplicación utilizaremos una computadora con las siguientes características:

- Procesador: Intel Core I3 2350M 2.30 GHz
- Memoria: 4096 MB RAM
- Disco Duro: 320 GB
- Sistema Operativo: Windows 7 Profesional 64 bits

5.5.2 Materiales Software

En el desarrollo de la aplicación utilizaremos las siguientes herramientas informáticas.

- Microsoft Visual Studio 2013 (Paquete de lenguaje de programación)
- Microsoft ASP.NET (Framework web)
- Microsoft Sql Server 2012 (Sistema Gestor de Bases de datos)
- HTML5 (Lenguaje de marcado)
- Css3 (Hojas de estilos para las interfaces)
- JavaScript (Validación de entrada de información)
- Internet Information Server (Servidor web)

VI. Diseño de la aplicación

La creación de esta aplicación está basado en conseguir las metas de diseño (Simplicidad, Identidad, Consistencia y Robustez), en esta sección se explicaran los aspectos más relevantes del diseño y estructura de la aplicación web:

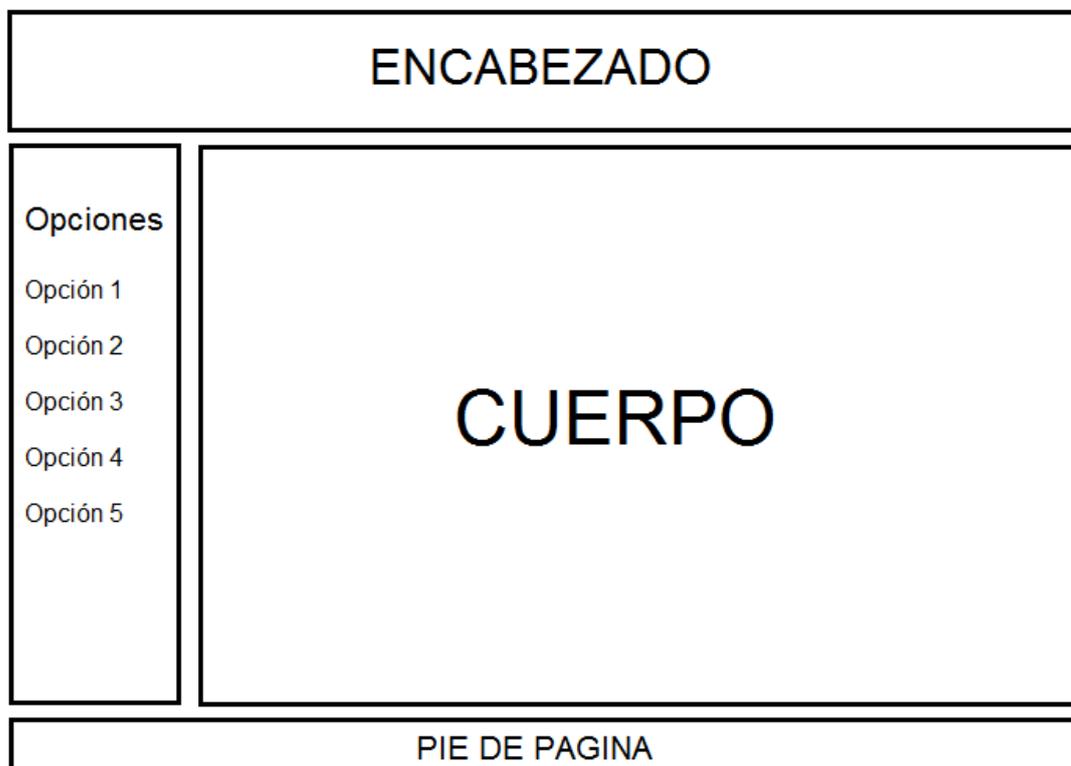
6.1 Especificaciones de usabilidad

La aplicación ha de ser suficientemente intuitiva para que pueda ser utilizada por usuarios sin conocimientos informáticos elevados. Para conseguir esto se ha optado por utilizar una interfaz gráfica simple, con botones que indican claramente su función y lo suficientemente descriptivos como para que el usuario

pueda saber intuitivamente para que sirven. Por otro lado, el hecho de las validaciones evita que el ingreso incorrecto de información a la aplicación pueda generar un problema a la hora de ejecución.

6.2 Diseño de las Páginas

Para la creación de todas las páginas de la aplicación web se utilizó una masterpage (Página Maestra) para que se heredara la misma estructura en todo el proyecto y de esta manera obtener una aplicación más amigable al usuario, Aprovechar la reutilización de código y facilitar el uso de la misma.



6.2.1 Opción Solicitud

Esta opción nos permite ingresar solicitudes para compra de materiales, productos, bienes o servicios que necesite el usuario solicitante, detallando cantidad, unidad de medida, descripción, especificaciones técnicas y monto reservado, además de estos datos se tomará del sistema, la fecha y hora de creación que nos permitirá controlar los tiempos en las diferentes etapas de la solicitud.

6.2.2 Opción Presupuesto

Esta opción permite que el área de presupuesto revise la solicitud de compra y aprobarla después de verificar la disponibilidad presupuestaria y la reserva correspondiente para asegurar el pago de la compra.

6.2.3 Opción Usuario

Esta opción permite al usuario administrador de la aplicación registrar nuevos usuarios, ingresando nombre, apellidos, contraseña, nombre de usuario, perfil, Facultad, Dependencia a la que pertenece.

6.2.4 Opción Proveedor

Esta opción permite al usuario administrador de la aplicación registrar nuevos proveedores, ingresando razón social, nombre comercial, número RUC, Correo electrónico, dirección, teléfono, Departamento y municipio al que pertenece.

6.2.5 Opción Orden de Compra

Esta opción permite al usuario de Adquisiciones generar de una solicitud su correspondiente orden de compra ingresando el proveedor, información del proceso de compra, costo de los productos, forma de pago, lugar y fecha de entrega.

6.2.6 Opciones Facultad, Dependencias, Fondos

Esta opción al administrador del sistema registrar las tablas complementarias de la aplicación, ingresando un nombre de cada tabla.

6.2.7 Opción Cambiar Contraseña

Esta opción permite al usuario realizar un cambio de contraseña método usado ingresar contraseña actual, ingresar nueva contraseña y confirmar contraseña nueva.

6.3 Estructura de la Aplicación:

6.3.1 Página Index.aspx

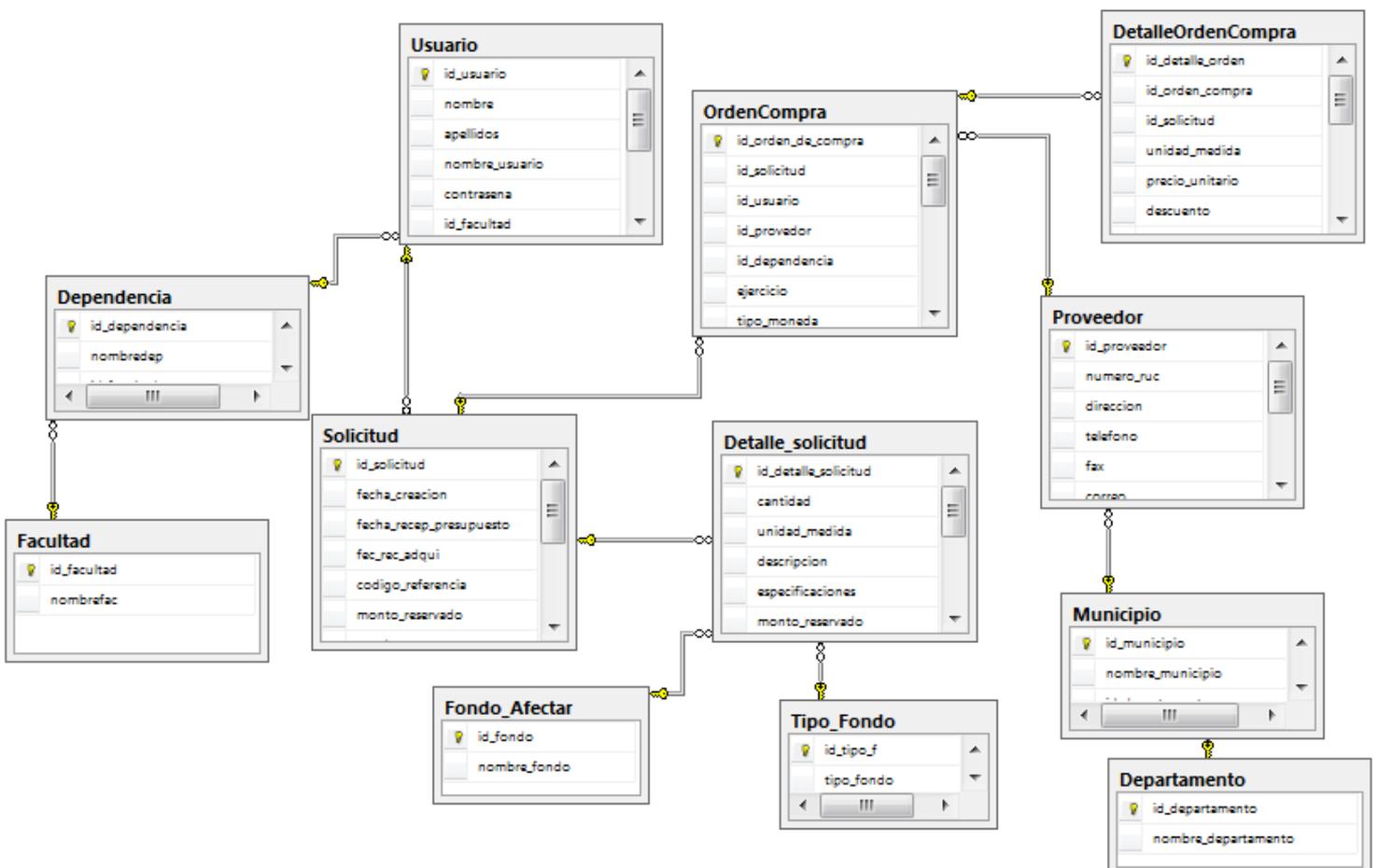
Esta es la página de inicio de la aplicación donde el usuario podrá ingresar sus datos para validar el inicio de sesión y cargar la página principal de la aplicación.

6.3.2 Página Principal.aspx

Esta es la página principal de la aplicación desde aquí se tiene acceso a todas las opciones de la aplicación.

6.3.3 Diseño de la base de datos

La aplicación web utiliza una base de datos llamada **Compras** creada en el sistema gestor de bases de datos Sql Server 2012, aquí se crearon las tablas necesarias para almacenar toda la información de nuestra aplicación. El diagrama entidad relación de esta base de datos es el siguiente:



Las tablas que componen la base de datos son las siguientes

6.3.3.1 Tabla Usuario

En esta tabla se almacena la información correspondiente a los usuarios de la aplicación web, los campos de la tabla son Id_usuario, nombre, apellidos, nombre_usuario, contraseña, id_facultad, id_dependencia, perfil, estado.

	Nombre de columna	Tipo de datos
	id_usuario	int
	nombre	varchar(50)
	apellidos	varchar(50)
	nombre_usuario	varchar(50)
	contrasena	nchar(50)
	id_facultad	int
	id_dependencia	int
	perfil	varchar(50)
	estado	varchar(50)

6.3.3.2 Tabla Proveedor

En esta tabla se almacena la información correspondiente a los proveedores que suministran los materiales, productos, bienes o servicios a la institución, los campos de la tabla son Id_proveedor, razón_social, nombre_comercial, numero_ruc, email, direccion, teléfono, departamento, municipio.

	Nombre de columna	Tipo de datos
	id_proveedor	int
	numero_ruc	varchar(14)
	direccion	varchar(500)
	telefono	varchar(20)
	fax	varchar(20)
	correo	varchar(50)
	razon_social	varchar(100)
	nombre_comercial	varchar(100)
	id_departamento	int
	id_municipio	int

6.3.3.3 Tabla Solicitud

En esta tabla se almacena la información correspondiente a la solicitud de compra, los campos de la tabla son Id_solicitud, fecha_creacion, fecha_recep_presupuesto, fec_rec_adqui, codigo_referencia, monto_reservado, estado, id_usuario, id_facultad, id_dependencia.

	Nombre de columna	Tipo de datos
	id_solicitud	int
	fecha_creacion	datetime
	fecha_recep_presupuesto	datetime
	fec_rec_adqui	datetime
	codigo_referencia	int
	monto_reservado	float
	estado	varchar(50)
	id_usuario	int
	id_facultad	int
	id_dependencia	int

6.3.3.4 Tabla Detalle_solicitud

En esta tabla se almacena el detalle de los materiales, productos, bienes o servicios a adquirir, los campos son Id_detalle_solicitud, cantidad, unidad_medida, descripción, especificaciones, monto_reservado, id_fondo, organica, ordinal, id_solicitud.

	Nombre de columna	Tipo de datos
	id_detalle_solicitud	int
	cantidad	int
	unidad_medida	varchar(20)
	descripcion	varchar(MAX)
	especificaciones	varchar(MAX)
	monto_reservado	float
	id_fondo	int
	organica	varchar(50)
	ordinal	varchar(50)
	id_solicitud	int

6.3.3.5 Tabla OrdenCompra

En esta tabla se almacena la información una vez generada la orden de compra de la solicitud de compra los campos de la tabla son: Id_orden_de_compra, id_solicitud, id_usuario, id_proveedor, id_dependencia, ejercicio, tipo_moneda, tipo_cambio, tipo_contratacion, norma_aplicable, numero_expediente, numero_resolucion, numero_acta, observacion, tipo_entrega, fecha

	Nombre de columna	Tipo de datos
▶ 🔑	id_orden_de_compra	int
	id_solicitud	int
	id_usuario	int
	id_proveedor	int
	id_dependencia	int
	ejercicio	int
	tipo_moneda	varchar(50)
	tipo_cambio	varchar(50)
	tipo_contratacion	varchar(50)
	norma_aplicable	varchar(50)
	numero_expediente	int
	numero_resolucion	varchar(10)
	numero_acta	varchar(10)
	observacion	varchar(200)
	tipo_entrega	varchar(50)
	fecha	datetime

6.3.3.6 Tabla DetalleOrdenCompra

En esta tabla se almacena el detalle de la información una vez generada la orden de compra de la solicitud de compra los campos de la tabla son: id_detalle_orden, id_orden_compra, id_solicitud, cantidad, unidad_medida, precio_unitario, descuento, precio_total_s_iva, exento_iva.

	Nombre de columna	Tipo de datos
▶ 🔑	id_detalle_orden	int
	id_orden_compra	int
	id_solicitud	int
	cantidad	int
	unidad_medida	varchar(50)
	precio_unitario	float
	descuento	float
	precio_total_s_iva	float
	exento_iva	varchar(50)

6.3.3.7 Tabla Facultad

En esta tabla se almacena la información correspondiente a las facultades que conforman la Institución, los campos de la tabla son Id_facultad, nombrefac.

	Nombre de columna	Tipo de datos
▶ 🔑	id_facultad	int
	nombrefac	varchar(100)

6.3.3.8 Tabla Dependencia

En esta tabla se almacena la información correspondiente a las dependencias que conforman la Institución, los campos de la tabla son Id_dependencia, nombredep, id_facultad.

	Nombre de columna	Tipo de datos
▶ 🔑	id_dependencia	int
	nombredep	varchar(100)
	id_facultad	int

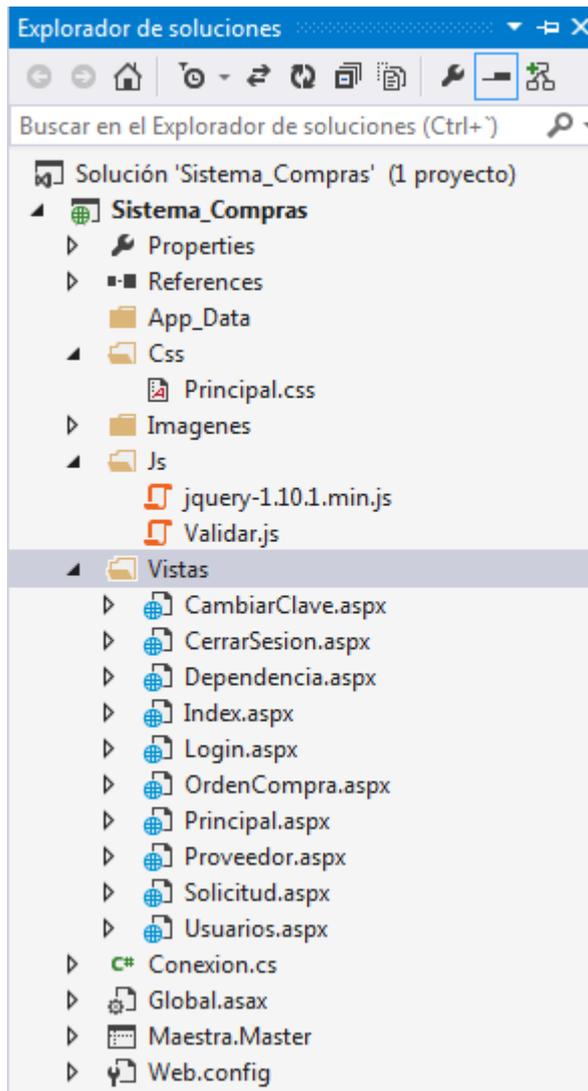
6.3.3.9 Script de creación de una de las tablas de la bases de datos, Tabla Proveedor

```
CREATE TABLE [dbo].[Proveedor](
    [id_proveedor] [int] NOT NULL,
    [numero_ruc] [varchar](14) NOT NULL,
    [direccion] [varchar](500) NOT NULL,
    [telefono] [varchar](20) NOT NULL,
    [fax] [varchar](20) NOT NULL,
    [correo] [varchar](50) NOT NULL,
    [razon_social] [varchar](100) NOT NULL,
    [nombre_comercial] [varchar](100) NOT NULL,
    [id_departamento] [int] NOT NULL,
    [id_municipio] [int] NOT NULL,
    CONSTRAINT [PK_Proveedor] PRIMARY KEY CLUSTERED
    (
        [id_proveedor] ASC
    )
)
```

VII. Implementación de la aplicación

7.1 Estructura de la aplicación

En esta fase del proyecto se ha obtenido el código de la aplicación web a partir de la documentación obtenida durante las fases de análisis y diseño. La aplicación se ha realizado separando los Web Forms, archivos Css, Archivos Javascript y clases, se crearon carpetas contenedoras para organizar los archivos y obtener una mejor apariencia, los Web Forms están guardados en archivos .aspx, la configuración de la apariencia guardada en archivo .Css, las validaciones almacenadas en archivo .Js, las clases funcionales guardadas en archivos .cs.



7.2 Codificación de la Conexión a la base de datos Compras

Quizá uno de los aspectos más interesantes de los utilizados en el desarrollo de la aplicación es el acceso a la base de datos desarrollada en Microsoft SQL Server 2012 mediante el control `SqlDataSource`. A continuación se muestra a modo de ejemplo, parte del código usado en la aplicación para este propósito:

```

Conexion.cs*  X
Sistema_Compras.Conexion  Conectar()
namespace Sistema_Compras
{
    21 referencias
    public class Conexion
    {
        //declaramos las variables de conexion
        public SqlConnection con;
        public DataSet ds = new DataSet();
        public SqlDataAdapter da;
        public SqlCommand comando;
        public DataTable dt;

        //Metodo que manda a llamar a la cadena de conexion
        1 referencia
        public void Conectar()
        {
            string cadena = ConfigurationManager.ConnectionStrings["Connection"].ConnectionString;
            con = new SqlConnection(cadena);
        }

        //Metodo Constructor
        10 referencias
        public Conexion()
        {
            Conectar();
        }
    }
}

```

En el directorio raíz de nuestra aplicación asp.net el archivo web.config en la configuración agregamos un elemento ConnectionString a como se muestra en la imagen siguiente.

```

Web.config*  X
<?xml version="1.0" encoding="utf-8"?>
<!--
Para obtener más información sobre cómo configurar la aplicación de ASP.NET, visite
http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
  </system.web>

  <connectionStrings>
    <add connectionString="Data Source=BAYARDO-PC; initial catalog=Compras; integrated security=true"
      name="Connection" providerName="System.Data.SqlClient" />
  </connectionStrings>
</configuration>

```

7.3 Herramientas Externas

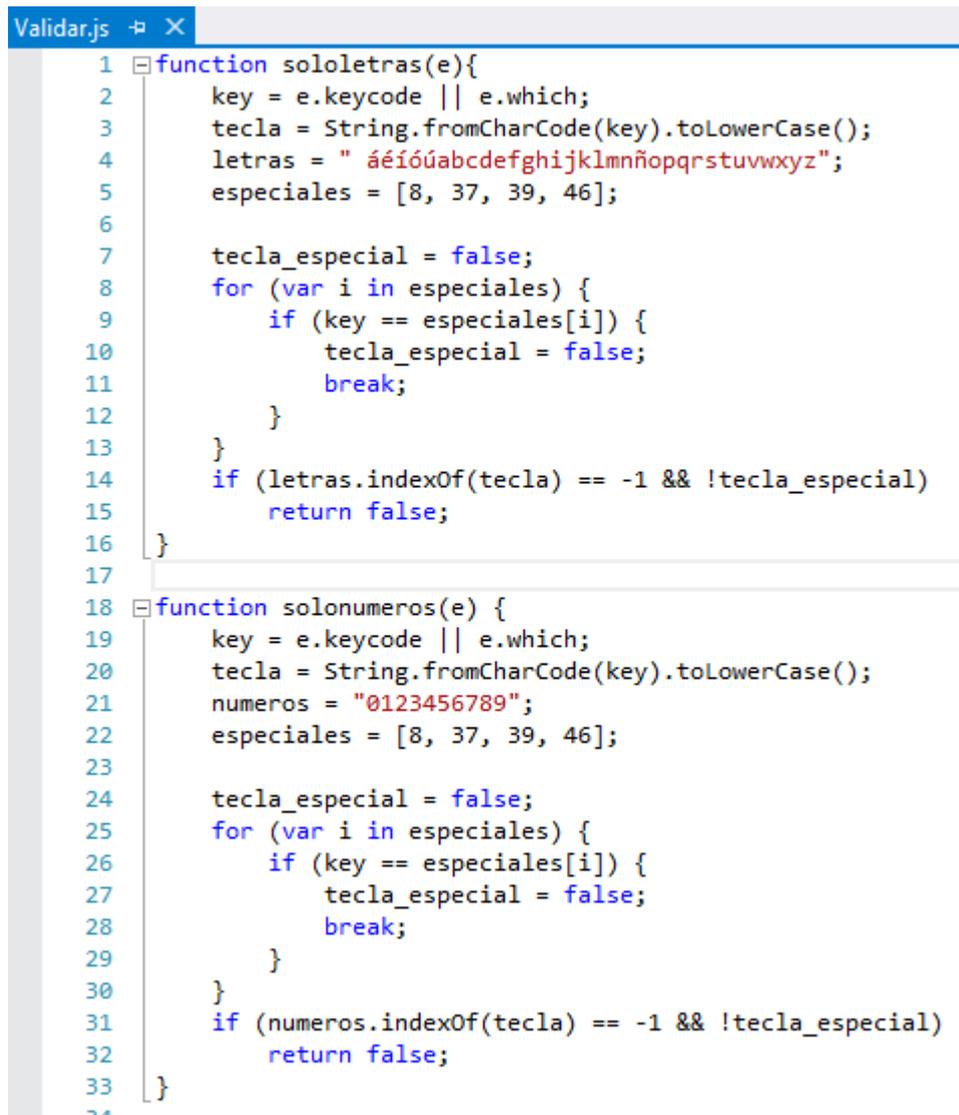
7.3.1 Uso de JavaScript

Se utilizó el lenguaje JavaScript para realizar las validaciones en nuestra aplicación, al momento de ingresar la información, para lograr esto se referencia el archivo

Validar.js desde nuestra página maestra para que esté disponible en toda nuestra aplicación, con estas instrucciones:

```
<script type="text/javascript" src="../../Js/jquery-1.10.1.min.js"></script>
<script type="text/javascript" src="../../Js/Validar.js"></script>
```

El primer archivo JQuery-1-10.1.min.js es una librería JQuery necesaria para ejecutar archivos JavaScript, El segundo archivo Validar.js es donde están las funciones de las validaciones, el contenido del archivo es el siguiente.



```
Validar.js  ▸ ×
1  function sololetras(e){
2      key = e.keycode || e.which;
3      tecla = String.fromCharCode(key).toLowerCase();
4      letras = " áéíóúabcdefghijklmnñopqrstuvwxyz";
5      especiales = [8, 37, 39, 46];
6
7      tecla_especial = false;
8      for (var i in especiales) {
9          if (key == especiales[i]) {
10             tecla_especial = false;
11             break;
12         }
13     }
14     if (letras.indexOf(tecla) == -1 && !tecla_especial)
15         return false;
16 }
17
18 function solonumeros(e) {
19     key = e.keycode || e.which;
20     tecla = String.fromCharCode(key).toLowerCase();
21     numeros = "0123456789";
22     especiales = [8, 37, 39, 46];
23
24     tecla_especial = false;
25     for (var i in especiales) {
26         if (key == especiales[i]) {
27             tecla_especial = false;
28             break;
29         }
30     }
31     if (numeros.indexOf(tecla) == -1 && !tecla_especial)
32         return false;
33 }
34
```

7.3.2 Uso de Css

Se utilizó el lenguaje Css para la presentación de las interfaces, para lograr esto se referencia el archivo Principal.css desde nuestra página maestra para que esté disponible en toda nuestra aplicación, con la siguiente instrucción:

```
<link href="Css/Principal.css" rel="stylesheet" type="text/css"/>
```

7.4 Seguridad en la Aplicación

En el desarrollo de la aplicación se utilizaron los siguientes mecanismos de seguridad

7.4.1 Autenticación de usuario al ingresar a la aplicación

Esto garantiza que solo los usuarios registrados con un usuario y clave autorizada puedan ingresar a la aplicación.

7.4.2 Validación de las variables de sesión

Se realizó validación con las variables de sesión para evitar que directamente desde la URL se pueda ingresar escribiendo la dirección de una página específica, al momento de ingresar a la aplicación se guarda la variable de sesión, de forma que si el usuario no ha ingresado vía autenticación la variable de sesión estará vacía.

7.4.2 Encriptación de clave

Se realizó encriptación utilizando las funciones `pwdencrypt` y `pwdcompare` para que la clave almacenada en la base de datos este encriptada y que de igual manera viaje a través de la red.

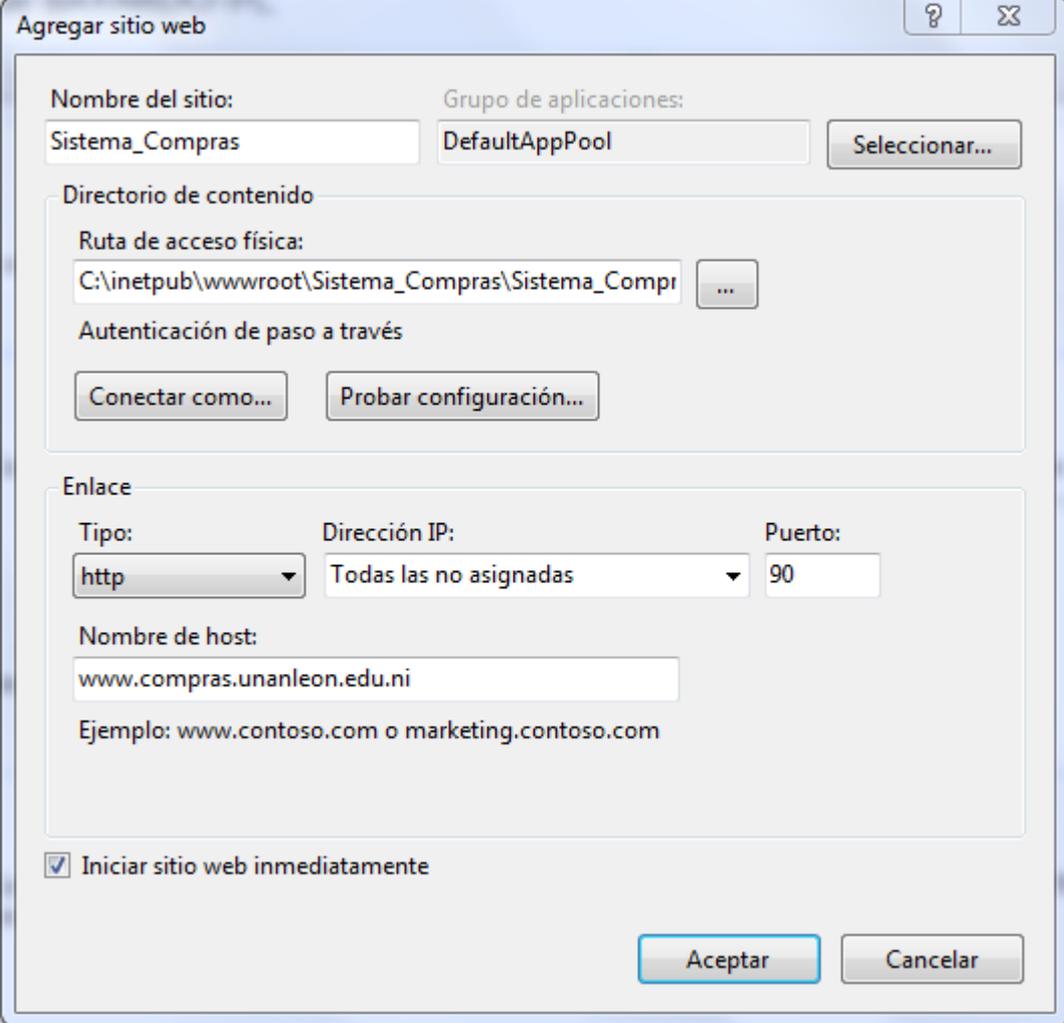
VIII. Publicación

8.1 Publicar aplicación web en Internet Information Server

IIS es nuestro administrador de sitios web donde se publican las aplicaciones web para que estén disponibles y se pueda tener acceso a través de Internet, en nuestro proyecto acceso a través del localhost, ya que tenemos instalado el IIS en nuestro equipo.

Se debe copiar la carpeta de la aplicación web en el directorio wwwroot ubicado en la ruta: C:\inetpub\wwwroot, este directorio se crea momento de instalar el IIS.

En el Administrador de Internet Information Services, lo primero que hacemos es agregar un sitio web especificando en los cuadros de diálogos nombre del sitio web, Grupo de aplicaciones, ruta de acceso física, protocolo para el sitio, dirección IP, puerto y host.



The screenshot shows the 'Agregar sitio web' (Add Website) dialog box in Internet Information Services (IIS) Manager. The dialog is titled 'Agregar sitio web' and contains the following fields and options:

- Nombre del sitio:** Sistema_Compras
- Grupo de aplicaciones:** DefaultAppPool (with a 'Seleccionar...' button)
- Directorio de contenido:**
 - Ruta de acceso física:** C:\inetpub\wwwroot\Sistema_Compras\Sistema_Compr (with a browse button '...')
 - Autenticación de paso a través:** (with 'Conectar como...' and 'Probar configuración...' buttons)
- Enlace:**
 - Tipo:** http (dropdown menu)
 - Dirección IP:** Todas las no asignadas (dropdown menu)
 - Puerto:** 90 (text box)
 - Nombre de host:** www.compras.unanleon.edu.ni (text box)
 - Ejemplo:** www.contoso.com o marketing.contoso.com
- Iniciar sitio web inmediatamente**
- Buttons:** Aceptar (Accept) and Cancelar (Cancel)

IX Conclusiones y trabajos futuros

9.1 Conclusiones

Con el proyecto de tesis finalizado llegamos a las siguientes conclusiones:

La aplicación web desarrollada ofrece a las áreas involucradas en los procesos de compras una herramienta que le permitirá ingresar sus solicitudes de compras y sobre todo poder dar seguimiento para ver el avance de dicha compra.

El uso de Sql Server 2012 como gestor de bases de datos y ASP.NET nos ha brindado una excelente compatibilidad entre estas herramientas y nos permitió el buen desarrollo de la aplicación de manera sencilla y rápida.

Se logró desarrollar una interfaz amigable al usuario, fácil de utilizar y con el mismo diseño en todas sus páginas para mantener el mismo esquema en toda la aplicación.

El resultado final del trabajo ha sido realmente muy enriquecedor ya que nos permitió tener una primera aproximación y entendimiento de la tecnología .NET de Microsoft, adquiriendo suficientes conocimientos como para desarrollar una aplicación Web de complejidad media. También ha servido como repaso y puesta en práctica de numerosos conceptos alcanzados en diversos módulos a lo largo

del master, especialmente de las asignaturas de desarrollo web, bases de datos y seguridad web.

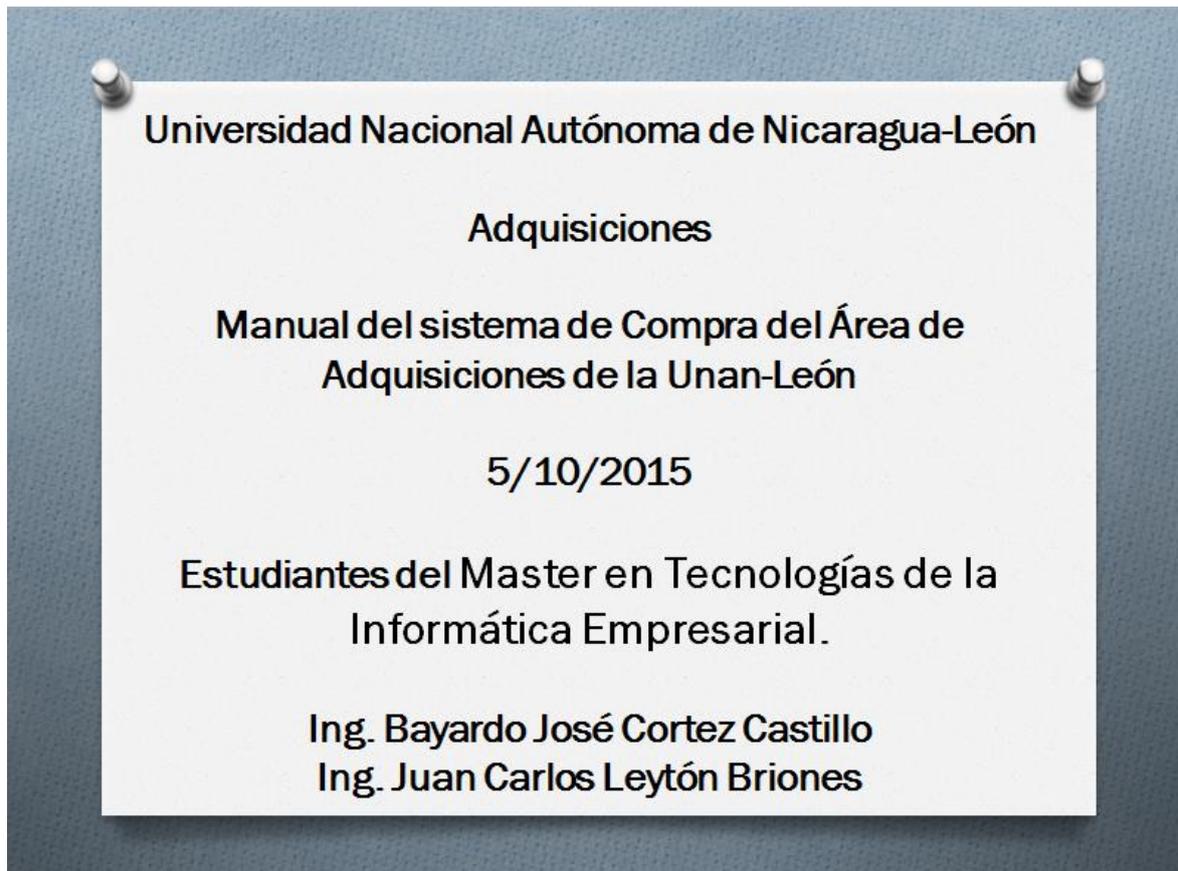
Se puede decir que se han alcanzado los objetivos propuestos inicialmente, tanto por el desarrollo de la aplicación como por los conocimientos adquiridos.

9.2 Líneas de trabajos futuros

- Culminar el modulo correspondiente a la generación de orden de compra.
- Agregar a la aplicación la funcionalidad de realizar respaldo de la base de datos de forma periódica.
- Agregar la funcionalidad de generación de reportes e informes.

ANEXOS

X. Manual de la aplicación



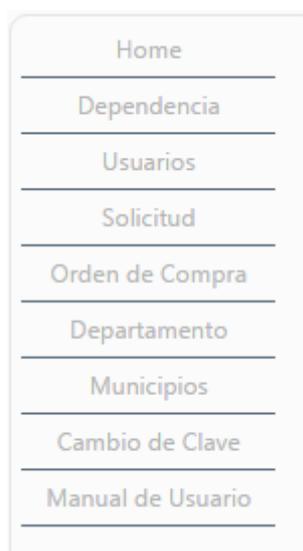
10.1 Ingreso al sistema

El sistema será accedido vía web a través de cualquier navegador y computadora dentro de la red de la UNAN-LEÓN Una vez que tengamos

acceso a la página principal de la aplicación web, nos aparecerá el siguiente formulario de Credenciales de usuario:



Aquí cada Operador del sistema deberá ingresar el usuario y contraseña asignado por el administrador del sistema, dependiendo de las credenciales de usuario, podrá ser administrador del sistema o tan solo un operador común del sistema, como administrador podrá ingresar datos de configuraciones tales como ingresar las Facultades y Dependencias de la Universidad, para ello tiene que seleccionar la opción correspondiente al menú del panel izquierdo:



10.2 Ingresar Facultad y Dependencia.

Luego de darle clic nos aparecerá el siguiente formulario donde tenemos que darle clic en el Radio Button Facultad y darle en el botón Elegir.

The screenshot shows a web application interface with two main sections. On the left, there is a form titled "Ingresar Facultad" (Enter Faculty). It contains a label "Facultad" above a text input field containing "Ciencias y Tecnologia". Below the input field are two buttons: "Actualizar" (Update) and "Cancelar" (Cancel). On the right, there is a table with the following data:

Editar	id_facultad	nombrefac
	1	Edificio Central
	2	Ciencias y Tecnologia
	3	Ciencias Medicina
	4	Humanidades
	5	Derecho
	6	Economia

Below the table, there is a search bar with a magnifying glass icon, a radio button labeled "Dependencia", and a button labeled "Elegir". The number "12" is displayed below the search bar.

Luego deberá ingresar nombre de la Facultad aquí tenemos validaciones del campo Facultad, donde deben de introducir solo caracteres y luego darle en el botón Guardar, ahí mismo en ese link podrá ingresar Dependencia, solo hay que darle clic en el Radio Button Dependencia y darle en el botón Elegir.

Ingresar Dependencia

Dependencia

Facultad

Editar	id_dependencia	nombredep	id_facultad	nombrefac
	1	Informatica	1	Edificio Central
	51	Dpt Computacion	2	Ciencias y Tecnologia
	52	Laboratorio	2	Ciencias y Tecnologia
	53	vip	1	Edificio Central
	54	VRA	1	Edificio Central
	55	VRE	1	Edificio Central

Dependencia
 Facultad

Aquí deberá ingresar el nombre de la Dependencia y la Facultad a la que pertenece, luego darle clic en el botón Guardar.

De igual forma podrá actualizar o borrar dependencia si así lo desea. Solo tiene q darle clic en la imagen de la edición y cambiar los nombres y darle clic en el Botón Actualizar.

Ingresar Dependencia

Dependencia

Facultad

Editar	id_dependencia	nombredep	id_facultad	nombrefac
	1	Informatica	1	Edificio Central
	51	Dpt Computacion	2	Ciencias y Tecnologia
	52	Laboratorio	2	Ciencias y Tecnologia
	53	vip	1	Edificio Central
	54	VRA	1	Edificio Central
	55	VRE	1	Edificio Central

Dependencia
 Facultad

10.4 Ingresar Usuario

Ingresar, Editar, Borrar y resetear contraseñas de usuarios, nos vamos al Menú principal le damos en el link de Usuario.



Dar Clic en Usuario nos aparecerá el siguiente formulario para ingresar nuevos usuarios.

Ingresar Usuarios

Nombres

Apellidos

Usuario

Contraseña

Facultad
 Ciencias y Tecnología

Dependencia
 Dpt Computacion

Perfil

Estado

		1	Juan Carlos	Leyton	jbriones		1	Edificio Central	1	Informatica	admon
		19	jose	briones	jb		1	Edificio Central	1	Informatica	operador
		51	carlos	briones	jc		1	Edificio Central	1	Informatica	prueba
		53	Jose Carlos	Lopez	Lopez		1	Edificio Central	54	VRA	Operado
		54	Denis Jose	Reyes	Reyes		4	Humanidades	52	Laboratorio	Operado

En este formulario debemos de ingresar los siguientes datos como son Nombres, Apellidos, Nombre de Usuario, Contraseña, Facultad, Dependencia, Perfil y Estado, luego darle clic en el botón Guardar. También el administrador del sistema podrá actualizar los datos de usuarios a cómo podrá resetear la contraseña de aquel usuario que no se acuerde de su contraseña.

Ingresar Usuarios

Nombres

Apellidos

Usuario

Contraseña

Facultad

Dependencia

Perfil

Estado

1	Juan Carlos	Leyton	jbriones	Informatica	Edificio Central	1	admon	activo
19	jose	briones	jb	Informatica	Edificio Central	1	operador	activo
51	carlos	briones	jc	Informatica	Edificio Central	1	prueba	prueba
53	Jose Carlos	Lopez	Lopez	VRA	Edificio Central	54	Operador	Inactivo
54	Denis Jose	Reyes	Reyes	Laboratorio	Humanidades	52	Operador	Activo

10.5 Cambio de clave de Usuario

Dar clic en menú principal donde dice cambio de clave.



Nos aparecerá el siguiente formulario, donde tenemos que introducir la clave actual y luego la clave nueva además repetir la misma clave nueva. Si la clave es verdadera y la clave nueva coincide con la clave repetida saldrá el mensaje cambio de clave satisfactoriamente.

Gestion de **Compra Unan-León**

Cambiar Clave del Usuario

Ingrese su Clave Actual:

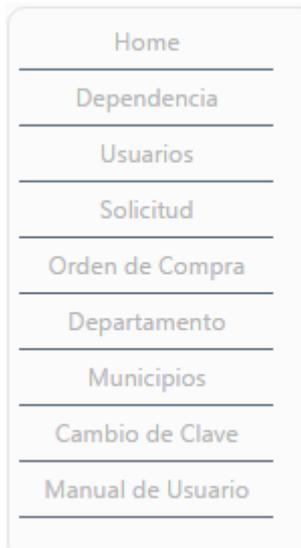
Recuerde anotar en un lugar seguro su nueva clave

Ingrese su Nueva Clave:

Repita su Nueva Clave:

10.3 Ingresar Departamentos y Municipios:

Podrá también ingresar todos los Departamentos y municipios del país, para insertar estos datos a la bases de datos tendrá que darle clic al menú principal.



Luego dar clic en el botón Departamento nos aparecerá el siguiente formulario.

Formulario de ingreso de departamentos:

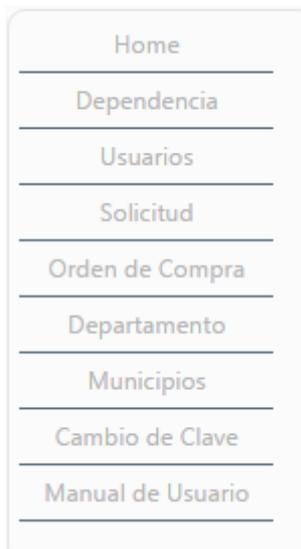
Ingresar Departamento

Departamento:

Editar	Eliminar	id_departamento	nombre_departamento
		1	Esteli
		2	León
		3	Managua
		4	Chinandega
		5	Jinotega
		6	Rivas

Buscador:

Donde tendremos que editar el nombre de Departamento y luego darle en el Botón Guardar, así mismo podremos ingresar los municipios, damos clic en el Botón Municipio del Menú.



En seguida nos aparecerá el siguiente formulario, donde debemos de ingresar el nombre del Departamento y luego el Nombre Municipio. También en el formulario Departamento como de municipios podremos editarlos.

Ingresar Municipios

Departamento: León

Municipio: La Paz Centro

Guardar Cancelar

Editar	Eliminar	id_municipio	nombre_municipio	id_departamento	nombre_departamento
		2	La paz Centro	1	Esteli
		3	chichigalpa	4	Chinandega
		4	Nagarote	3	Managua
		5	Rivas	6	Rivas

10.6 Administradores del Sistema:

Para consultas y resolución de problemas sobre el sistema de compra de nuestra Universidad debe ponerse en contacto con los desarrolladores de la aplicación.

Ing. Bayardo José Cortez Castillo

Ext. 1028

bayardo.cortez@ac.unanleon.edu.ni

Ing. Juan Carlos Leytón Briones

Ext. 1038

jbriones@di.unanleon.edu.ni

XI. Bibliografía

- Guerrero, M. F. (s.f.). *Microsoft*. Recuperado el 5 de Marzo de 2015, de <https://msdn.microsoft.com/es-es/library/bb972208.aspx>
- <https://es.scribd.com>. (05 de Septiembre de 2015). Obtenido de <https://es.scribd.com/doc/56767327/Definicion-de-Sevidor-IIS>
- Jimenez, M. (12 de Octubre de 2013). *Prezy*. Recuperado el 6 de Marzo de 2015, de <https://prezi.com/rax0yyqrkv0i/modelo-en-cascada/>
- LexisNexis. (2012). *ASP.NET 4.5 en C# con Visual Studio 2012*.
- Microsoft*. (30 de Septiembre de 2015). Obtenido de <https://msdn.microsoft.com/es-es/library/4w3ex9c2%28v=vs.100%29.aspx>
- Mozilla*. (05 de Septiembre de 2015). Obtenido de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Pereira, J. E. (1 de Diciembre de 2006). *GestioPolis*. Recuperado el 4 de Marzo de 2015, de <http://www.gestiopolis.com/>
- Prezy.com*. (05 de Septiembre de 2015). Obtenido de <https://prezi.com/absznnvrqpsr/historia-y-definicion-de-microsoft-sql-server/>
- Stallman, R. (31 de Octubre de 2011). *Sociología y Redes Sociales*. Recuperado el 10 de Julio de 2015, de <http://sociologiayredessociales.com/2011/10/richard-stallman-software-libre-libertad-en-la-red/>