UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN-LEÓN

FACULTAD DE CIENCIAS Y TECNOLOGÍA

DEPARTAMENTO DE COMPUTACIÓN

TESIS PARA OPTAR AL TITULO DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN.

Desarrollo de un sistema Web en ASP.net para la gestión de recursos humanos basada en nóminas para la empresa ACAPROF ubicada en Matagalpa Nicaragua durante el período Enero a Mayo de 2019

Autores:

- Br. Erving José Salazar Tejeda.
- Br. Franklin Antonio Altamirano Arbizu.
- Br. Luis Manuel Oviedo Acosta.

Tutor: Ing. Juan Carlos Antón Soto.

¡A la libertad por la Universidad!

AGRADECIMIENTOS

A DIOS por el don de la vida y la inspiración de nuestro espíritu, lo que nos permitió culminar esta tesis,

A mis padres, porque con sus valores y consejos se convirtieron en un apoyo y motivación constante,

A la Universidad Nacional Autónoma de Nicaragua (UNAN – León), especialmente a su carrera de Ingeniería en Sistemas de Información a la cual muy orgullosamente pertenecemos,

A los docentes de cada materia impartida desde el inicio de nuestros estudios, porque con su conocimiento y experiencia aportaron en nuestra formación profesional,

A todas aquellas personas, que de una u otra forma aportaron a que nuestro sueño de ser profesionales se cumpla,

Y de manera muy especial al ingeniero Juan Calos Antón Soto, por su tiempo compartido, su apoyo incondicional y por impulsar el desarrollo de nuestra formación profesional.

Erving José Salazar Tejeda

En primer lugar, agradezco a Dios por Haberme permitido lograr y cumplir una de mis grandes metas y permitirme tener a mis padres, hermanos y familiares los cuales disfrutarán conmigo este gran logro.

En un lugar muy destacado quiero agradecer a mis padres ya que han sido el eje principal para la culminación de esta etapa, sin su apoyo esto no sería posible, principalmente a mi madre ROSA AMELIA ARBIZU por enseñarme a nunca bajar los brazos para cumplir cada una de mis objetivos, a no derrumbarme en ningún momento y seguir adelante a pesar de las adversidades.

He de expresar mi profundo agradecimiento a aquellas personas que me han ayudado a realizarme tanto intelectual como personalmente ya que su ayuda ha sido de gran importancia para poder llegar al sitio que me encuentro.

A la Universidad Nacional Autónoma de Nicaragua (UNAN – León), especialmente a su carrera de Ingeniería en Sistemas de Información a la cual muy orgullosamente pertenezco.

Este proyecto es el cierre de una puerta para abrirse otra, sin dejar de mencionar a aquellas personas que lo han hecho posible, ya que el camino ha sido muy largo pero a la vez lleno de satisfacciones. Y por ende quiero dejar constancia de mi agradecimiento.

Y de manera muy especial al ingeniero Juan Calos Antón Soto, por su tiempo compartido, su apoyo incondicional y por impulsar el desarrollo de nuestra formación profesional.

Franklin Antonio Altamirano Arbizu.

Gracias a mis padres, que fueron mis mayores promotores durante este proceso,

Gracias a Dios, que fue mi principal apoyo y motivador para cada día continuar sin tirar la toalla.

Gracias a mi universidad, gracias por haberme permitido fórmame y en ella, gracias a todas las personas que fueron participes de este proceso, ya sea de manera directa o indirecta, gracias a todos ustedes, fueron ustedes los responsables de realizar su pequeño aporte, que el día de hoy se vería reflejado en la culminación de mi paso por la universidad.

Quiero agradecerle al Ing. Juan Carlos Antón por cada detalle y momento dedicado para aclarar cualquier tipo de duda que nos surgiera, agradecerle por la caridad y la exactitud de sus observaciones.

Este es un momento muy especial que espero, perduré en el tiempo, no solo en la mente de las personas a quienes agradecí, sino también a quienes invirtieron su tiempo para echarle una mirada a mi proyecto de tesis, a ellos asimismo les agradezco con todo mí ser.

Luis Manuel Oviedo Acosta

DEDICATORIA

A Dios, por darme la vida y fuerzas necesarias para seguir adelante, A todas aquellas personas que formaron parte de mi carrera profesional, A mis padres; Pablo Salazar, Georgina Tejeda que siempre estuvieron apoyándome incondicionalmente en cada momento de mi vida. Asimismo, a mi asesor por su gran contribución en la presente tesis.

Erving José Salazar Tejeda

A Dios por permitirme la vida y poder ser la persona que soy, por iluminar mi mente y estar conmigo en cada paso que doy.

Este trabajo lo dedico a mi Madre que con mucho esfuerzo y dedicación me ha apoyado a lo largo de este camino que fue la culminación de mis estudios y gracias al poner toda su confianza y lograr así que sea una excelente persona. Hoy más que nunca quiero decirle, de lo orgulloso que estoy de ser hijo de esta gran mujer: “ROSA AMELIA ARBIZU”.

A mis hermanos por darme ese apoyo que siempre lo necesité.

Franklin Antonio Altamirano Arbizu.

El presente trabajo investigativo lo dedico principalmente a Dios, por ser el inspirador y darnos fuerza para continuar en este proceso de obtener uno de los anhelos más deseados.

A mis padres, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí´ y convertirme en lo que soy. A todas las personas que nos han apoyado y han hecho que el trabajo se realice con éxito en especial a aquellos que nos abrieron las puertas y compartieron sus conocimientos.

Luis Manuel Oviedo Acosta.

CONTENIDO

1. INTRODUCCIÓN 12

2. ANTECEDENTES 13

3. JUSTIFICACIÓN 14

4. PLANTEAMIENTO DEL PROBLEMA 15

5. OBJETIVOS 16

5.1 Objetivo General 16

5.2 Objetivos Específicos 16

6. MARCO TEÓRICO 17

6.1 Microsoft Visual Studio 17

6.2 NET 18

6.2.1 ¿Qué es .NET? 18

6.3 ASP.NET 19

6.3.1 Historia 19

6.3.2 Características 21

6.3.3 Extensiones 28

6.3.4 Modelos de programación en ASP.NET 29

6.3.5 Uso actual del lenguaje 30

6.4 Bootstrap 31

6.5 JavaScript 31

6.5.1 Características 32

6.6 Hojas de estilo en cascada (CSS) 34

6.7 DevExpress 34

6.7.1 XtraChart 35

6.8 Microsoft SQL Server 37

6.8.1 Características de Microsoft SQL Server 37

6.8.2 Herramientas de SQL Server 38

6.8.3 Modelos de bases de datos 40

6.9 Encriptar 42

6.9.1 AES 42

6.9.2 MD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 44

6.10 Visual Basic 46

6.10.1 Historia 46

6.10.2 Características 47

6.10. 3 Entorno de desarrollo 48

6.10.4 Ventajas 50

6.10.5 Desventajas 51

6.11 Arquitectura Cliente-Servidor 51

7. DISEÑO METODOLÓGICO 53

7.1 Metodología del Trabajo 53

7.2 Recursos Disponibles y Necesarios 57

8. ESPECIFICACIÓN DE REQUISITOS SOFTWARE 58

8.1 Especificación de requisitos software según el estándar de IEEE-8330 58

8.1.1 Propósito 58

8.1.2 Alcance del sistema 58

8.1.3 Personal involucrado 60

8.2 Definiciones, Acrónimos y Abreviaturas 61

8.2.1 De la empresa 61

8.2.2 Del Sistema 62
8.2.3 De tecnología 63
8.3 Referencias 65
8.4 Descripción general 66
8.4.1 Perspectiva del producto. 66
8.4.2 Funcionalidad del producto. 66
8.4.3 Características de usuario. 67
8.5 Perfiles de usuarios 67
8.5.1 Jerarquía de usuarios 68
8.6 Restricciones. 69
8.6.1 Políticas reguladoras 69
8.6.2 Limitaciones de hardware 70
8.6.3 Interfaces con otras aplicaciones 70
8.6.4 Funciones de control 70
8.7 Requisitos específicos 70
8.7.1 Interfaces Externas. 70
8.7.2 Requerimientos Funcionales 71
8.7.3 Requerimientos no Funcionales. 76
8.8 Requisitos comunes de las interfaces 80
8.8.1 Interfaces de usuario 80
8.8.2 Interfaces de hardware 80
8.8.3 Interfaces de software 81
8.8.4 Interfaces de comunicación 81
8.9 Requisitos funcionales 81
8.9.1 Requisito funcional 1 81
8.9.2 Requisito funcional 2 82
8.9.3 Requisito funcional 3 83

8.9.4 Requisito funcional 4 84
8.9.5 Requisito funcional 5 85
8.9.6 Requisito funcional 6 85
8.9.7 Requisito funcional 7 88
8.9.8 Requisito funcional 8 90
8.9.9 Requisito funcional 9 91
8.9.10 Requisito funcional 10 92
8.9.11 Requisito funcional 11 92
8.10 Requisitos no funcionales 94
8.10.1 Requisitos de rendimiento 94
9. CASOS DE USOS 96
9.1 Diagrama de Casos de Uso 96
9.2 Diagrama de Secuencia 98
10. DIAGRAMA ENTIDAD RELACIÓN 104
11. DIAGRAMA DE CLASES 105
12. DIAGRAMA DE RELACIÓN 106
13. ANÁLISIS Y DISEÑO 107
13.1. Documentación de Tablas de Entidades 107
13.1.1 Estructura de tablas de la base de datos 107
13.2. Diagrama de Actividades 125
14. CRONOGRAMA 128
15. CONCLUSIONES 129
16. RECOMENDACIONES 130
16.1 Recomendaciones de uso o implementación 130

16.2 Mejoras o trabajos futuros. 130

17. BIBLIOGRAFÍA 131

18. ANEXOS 132

18.1 Introducción 132

18.1.1 ¿Cuál es el objetivo del Manual de Usuarios? 132

18.2 Requerimiento Básico. 132

18.3 Acceso a la aplicación 133

18.4 Roles. 134

18.5 Pantalla de bienvenida 134

18.6 Perfil administrador 135

18.6.1 Configuración básica requerida 135

18.6.2 Agregar empresa 139

18.6.3 Agregar perfil de empleado 140

18.6.4 Agregar usuario 141

18.7 Perfil ejecutivo de cuenta 142

18.7.1 Importar planillas de un Excel al sistema 142

18.7.2 Visualizar planillas 145

18.7.3 Reportes 145

18.8 Perfil Gerente 146

18.8.1 Reporte de boletas de pagos de empleados masivos por empresa 146

18.8.2 Aprobar o rechazar solicitud de vacaciones 147

18.8.3 Aprobar planillas 148

18.9 Perfil Empleado 149

18.9.1 Ver perfil 149

18.9.2 Ver Boletas de pago 149
18.9.3 Solicitar vacaciones 151
18.9.4 Reporte de deducciones 151
18.10. Conclusión 152
19. RECURSOS 153
19.1. Recursos Humanos 153
19.2. Recursos Físicos 153
19.3. Recursos Financieros 153

1. INTRODUCCIÓN

El presente siglo ha traído grandes cambios para las organizaciones, dado que hemos pasado de una economía basada en la manufactura a una economía basada en la información. Esto implica que las Organizaciones Modernas deben obtener herramientas para la toma de decisiones en tiempo y forma auxiliándose para tales efectos de sistemas de información basados en computadoras que les permitan agilizar sus operaciones y que se convierten día a día en herramientas de vital importancia para cualquier empresa que busque superación.

Es indispensable en la actualidad un Sistema que nos brinde el apoyo adecuado para la Administración de planillas y reportes de las boletas de pagos, este software permite tener eficiencia tanto en control, así como en el manejo de las nóminas de la empresa incorporada en el Sistema.

Este trabajo de investigación está organizado de manera secuencial, es decir, a medida que se vaya obteniendo información detallada por la empresa, se avanza continuamente hasta completar el trabajo de Investigación y cumplir con los objetivos y requerimientos propuestos en la elaboración del sistema y la automatización para dicha empresa.

Se utiliza la Metodología Orientada a Objetos (MOO), ya que ésta nos permite modelar de manera eficaz todos y cada uno de los componentes que posee el Sistema Final. Se analizan los requerimientos tecnológicos y se hace énfasis en el hecho de que la arquitectura tecnológica con la que cuenta la Empresa ACAPROF es suficiente para desarrollar el Sistema; tomando las características de dicho Sistema y las necesidades del cliente.

En el diseño del Sistema se identifican elementos claves para el buen desempeño y aceptación por parte de los usuarios, estos elementos se encuentran divididos en: Software, Hardware y Usuarios. Una vez identificados estos elementos dentro de la Empresa, se efectúa un análisis de las características que tienen que poseer, a fin de garantizar el cumplimiento eficiente de las tareas que dispone el Sistema.

2. ANTECEDENTES

La Academia de Capacitación y Asesoría Profesional fue creada el 29 de julio del año 2017, en la ciudad de Matagalpa por iniciativa de los dos miembros principales de la sociedad, esta surgió con el propósito de incorporar las nuevas tecnologías de la información y la comunicación en el proceso de enseñanza-aprendizaje para contribuir al desarrollo en la metodología de educación de nuestro país.

Dicha academia brinda asesorías en las áreas contables, administrativas y financieras hacia las empresas, permitiendo hoy en día, generar fuentes de empleos a jóvenes capacitados, promoviendo de esta forma el desarrollo continuo a los estudiantes nicaragüenses.

En marzo del año 2018 se apertura la primera convocatoria de cursos virtuales, logrando gran aceptación en el segmento de mercado (estudiantes universitarios y profesionales). Actualmente la Academia cuenta con seis miembros colaboradores para la realización de sus funciones, los que se caracterizan por tener alto espíritu emprendedor, integridad ética y profesional.

Actualmente la academia presta servicios contables, financieros y administrativos a diversos negocios, contando con un sistema para control de inventarios a través del software COUNTING, mismo que es utilizado por algunos de los negocios, de los cuales prestamos nuestros servicios, pero no cuenta con un sistema automatizado que le permita elaborar, gestionar y controlar las planillas de pagos de los negocios, contando solamente con herramientas automatizadas en Microsoft Excel, por lo que se considera de suma importancia la propuesta de un software que le permita realizar estas gestiones de manera más rápida y efectiva.

3. JUSTIFICACIÓN

El uso de nuevas tecnologías informáticas permite una interacción más asertiva entre las organizaciones y sus integrantes, debido al avance tecnológico y la necesidad de satisfacer a sus clientes de manera eficiente, sus expectativas de recibir servicios rápidos y fiables, cada negocio debe desarrollar metodologías del manejo de la información para estar al nivel de sus diferentes competidores. Además de ser muy necesario llevar un control de los datos de cada cliente para cuando sea necesario poder tener soporte y poder presentarlos.

ACAPROF actualmente procesa su información manualmente lo que le genera muchos inconvenientes; errores humanos, lentitud y demanda de mayor cantidad de personal. Esto muestra la necesidad de implementar un sistema de gestión de datos especializado que ayude a mejorar la funcionalidad del negocio.

La mayor parte de sus clientes se encuentran en posiciones geográficas diferentes al de las oficinas de la empresa, y el acceso a la información es de vital importancia. Con la implementación de este proyecto ACAPROF se beneficiará de poseer un sistema, como lo es una página web, dando una mayor accesibilidad de datos actuales, y en algunos casos en tiempo real referentes a las actividades inherentes a la empresa, fortaleciéndose e innovando la imagen del mismo en la llamada *era de la información*, permitiendo un mayor alcance y presencia en el mundo virtual.

Nuestro propósito es demostrar la utilidad de los sistemas software en los negocios, promover e incentivar su utilización, en base a resultados que puedan ser demostrados.

4. PLANTEAMIENTO DEL PROBLEMA

Actualmente las empresas poseen una inquietud en lo que respecta a la mejora del proceso de toma de decisiones, acciones correctivas y preventivas para el progreso de las gestiones internas, puesto que se entiende que de ello dependerá el éxito o fracaso de su gestión. En función a lo anterior se ha visto como en el transcurrir de las décadas las herramientas gerenciales se han transformado, siendo más innovadoras, eficientes y productivas. Para el logro de los objetivos y metas las organizaciones requieren controlar oportunamente la ejecución de sus actividades y los resultados de las gestiones de nóminas, es por ello que requieren de herramientas automatizadas que faciliten llevar los puntos de control de forma eficaz y le permita alcanzar los niveles de competitividad y desempeño deseados.

Para dar solución a la situación planteada, surge la necesidad de realizar un sistema de gestión de nóminas para la empresa ACAPROF donde se puedan gestionar y administrar las planillas de los empleados, se podrá llevar un historial de los datos de cada empleado, así como sus boletas de pagos, gestionar sus vacaciones y controlar quienes están de forma activa en la empresa.

5. OBJETIVOS

5.1 Objetivo General

Desarrollar un sistema Web en ASP.net que le permita a la empresa ACAPROF llevar un control digitalizado y ágil en el sistema de pago de planilla y administración de Recursos Humanos, obteniendo mayor precisión, seguridad y rapidez de la información.

5.2 Objetivos Específicos

- Recopilar la información necesaria que nos ayude a analizar los procesos requeridos para que el sistema de respuestas eficaces a los principales problemas.

- Diseñar Interfaces que permitan el rápido y fácil ingreso de los datos.

- Brindar a los usuarios una interfaz amigable que les permitan gestionar sus vacaciones e información de pagos.

- Permitir que los encargados de aprobar las planillas puedan generar diferentes tipos de reportes necesarios para validar su control contable.

- Utilizar Procedimientos Almacenados y Funciones por medio de SQL Server para obtener un rápido acceso a la base de datos.

- Desarrollar niveles de seguridad de acceso a los usuarios, que permitan al administrador decidir quiénes pueden tener accesos a los diferentes procesos presentes en el sistema.

6. MARCO TEÓRICO

En la realización del sistema para la empresa ACAPROF se empleará una serie de términos y herramientas las cuales hemos decidido conceptualizar para un mayor entendimiento de nuestro trabajo investigativo.

6.1 Microsoft Visual Studio

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, JAVA, Python, Ruby y PHP, al igual que entornos de desarrollo web, como ASP.NET MVC, Django, etc., a lo cual hay que sumarle las nuevas capacidades online bajo Windows Azure.

Visual Studio permite a los desarrolladores crear sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002).Así, se pueden crear aplicaciones que se comuniquen entre estaciones de trabajo, páginas web, dispositivos móviles, dispositivos embebidos y consolas entre otros.

Microsoft .NET es el conjunto de nuevas tecnologías en las que Microsoft ha estado trabajando durante los últimos años con el objetivo de obtener una plataforma sencilla y potente para distribuir el software en forma de servicios que puedan ser suministrados remotamente y que puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados.

6.2 NET

6.2.1 ¿Qué es .NET?

.NET es una parte integral de muchas aplicaciones que se ejecutan en Windows y proporciona funcionalidad común para aquellas aplicaciones que se ejecutan. Esta descarga es para la gente que necesita. NET para ejecutar una aplicación en su ordenador. Para los desarrolladores, el Marco. NET proporciona un modelo de programación completo y coherente para la creación de aplicaciones que tienen experiencias de usuario visualmente impresionantes y una comunicación segura y sin problemas.

Es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones. Basado en ella, la empresa intenta desarrollar una estrategia horizontal que integre todos sus productos, desde el sistema operativo hasta las herramientas de mercado.

.NET podría considerarse una respuesta de Microsoft al creciente mercado de los negocios en entornos Web, como competencia a la plataforma Java de Oracle Corporación y a los diversos framework de desarrollo web basados en PHP. Su propuesta es ofrecer una manera rápida y económica, a la vez que segura y robusta, de desarrollar aplicaciones o como la misma plataforma las denomina, soluciones, permitiendo una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo. El Framework. NET es un componente integral de Windows que admite la creación y ejecución de la próxima generación de aplicaciones y servicios Web XML... NET Framework está diseñado para cumplir los siguientes objetivos:

- Para proporcionar un consistente entorno orientado a objetos de programación si el código objeto se almacena y se ejecuta a nivel local, ejecutado a nivel local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca al mínimo la implementación de software y los conflictos de versiones.
- Proporcionar un entorno de ejecución de código que promueve la ejecución segura del código, incluyendo el código creado por un tercero desconocido o semi confianza.

- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos de secuencias de comandos o interpretado.

6.3 ASP.NET

ASP.NET es un entorno para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores y diseñadores para construir sitios web dinámicos, aplicaciones web y servicios web XML. Apareció en enero de 2002 con la versión 1.0 del .NET Framework, y es la tecnología sucesora de la tecnología Active Server Pages (ASP). ASP.NET está construido sobre el Common Language Runtime, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework.

6.3.1 Historia

Microsoft introdujo la tecnología llamada Active Server Pages en diciembre de 1996. Es parte de Internet Information Services (IIS) desde la versión 3.0 y es una tecnología de páginas activas que permite el uso de diferentes scripts y componentes en conjunto con el tradicional HTML para mostrar páginas generadas dinámicamente. La definición contextual de Microsoft es que "Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para la web".

Después del lanzamiento del Internet Information Services 4.0 en 1997, Microsoft comenzó a investigar las posibilidades para un nuevo modelo de aplicaciones web que pudiera resolver las quejas comunes sobre ASP, especialmente aquellas con respecto a la separación de la presentación y el contenido y ser capaz de escribir código "limpio". A Mark Anders, un administrador del equipo de IIS y Scott Guthrie, quien se había unido a Microsoft en 1997 después de graduarse de la Universidad Duke, se le dio la tarea de determinar cómo debería ser ese modelo. El diseño inicial fue desarrollado en el curso de dos meses por Anders y Guthrie, Guthrie codificó los prototipos iniciales durante las celebraciones navideñas de 1997.

El prototipo inicial fue llamado "XSP"; Guthrie explicó en una entrevista en el año 2007 que, "La gente siempre se preguntaba qué significaba la X. En ese momento, realmente no significaba nada. XML comenzaba así, al igual que XSLT. Todo lo novedoso parecía empezar con una X, así que ese es el motivo por el que originalmente lo llamamos así".

El desarrollo inicial de XSP fue hecho usando Java, pero pronto se decidió construir una nueva plataforma sobre el Common Language Runtime (CLR), pues ofrecía un ambiente orientado a objetos, recolección de basura y otras características que fueron vistas como características deseables. Guthrie describió esta decisión como un "alto riesgo", pues el éxito de su nueva plataforma de desarrollo web estaría atado al éxito del CLR, que, como XSP, aún estaba en etapas tempranas de desarrollo, tanto así que el equipo XSP fue el primer equipo en Microsoft en enfocarse en el CLR.

Con el cambio al Common Language Runtime, XSP fue implementado en C# (conocido internamente como "Project Cool" pero mantenido en secreto para el público), y fue renombrado a ASP+, en este punto la nueva plataforma fue vista como el sucesor de Active Server Pages, y la intención fue proporcionar un medio fácil de migración para los desarrolladores ASP.

La primera demostración pública y la liberación de la primera beta de ASP+ (y el resto del .NET Framework) se realizaron en el Microsoft Professional Developers Conference (PDC) el 11 de julio de 2000 en Orlando, Florida. Durante la presentación de Bill Gates, Fujitsu demostró ASP+ usado en conjunción con COBOL 5 y el soporte para una variedad de otros lenguajes fue anunciada, incluyendo los nuevos lenguajes de Microsoft, Visual Basic .NET y C#, así como también el soporte por medio de herramientas de interoperabilidad para Python y Perl creadas por la empresa canadiense ActiveState.

Una vez que la marca ".NET" fue seleccionada en la segunda mitad del 2000 se cambió el nombre de ASP+ a ASP.NET. Mark Anders explicó en una aparición en The MSDNShow en ese año, "La iniciativa .NET comprende un número de factores, trata sobre la entrega de software como servicio, sobre XML y servicios web y la mejora real del Internet en términos de qué puede hacer... de verdad queremos llevar su nombre (de ASP+) más alineado con el resto de las piezas de la plataforma que componen el .NET framework" Después de cuatro años de desarrollo, y una serie de versiones de evaluación en los años 2000 y 2001 , ASP.NET 1.0 fue liberado el 5 de enero de 2002 como parte de la versión 1.0 del .NET Framework. Incluso antes de su liberación, docenas de libros habían sido escritos sobre ASP.NET.

6.3.2 Características

6.3.2.1 Páginas.

Las páginas de ASP.NET, conocidas oficialmente como "web forms" (formularios web), son el principal medio de construcción para el desarrollo de aplicaciones web. Los formularios web están contenidos en archivos con una extensión ASPX; en jerga de programación, estos archivos típicamente contienen etiquetas HTML o XHTML estático, y también etiquetas definiendo Controles Web que se procesan del lado del servidor y Controles de Usuario donde los desarrolladores colocan todo el código estático y dinámico requerido por la página web.

Adicionalmente, el código dinámico que se ejecuta en el servidor puede ser colocado en una página dentro de un bloque <% -- código dinámico -- %> que es muy similar a otras tecnologías de desarrollo como PHP, JSP y ASP, pero esta práctica es, generalmente, desaconsejada excepto para propósitos de enlace de datos pues requiere más llamadas cuando se genera la página. ASP.NET no sólo funciona sobre el servidor de Microsoft IIS, también lo hace sobre Apache.

6.3.2.2 El modelo Code-behind.

Microsoft recomienda que para realizar programación dinámica se use el modelo code-behind, o de respaldo, que coloca el código en un archivo separado o en una etiqueta de script especialmente diseñada. Los nombres de los archivos code-behind están basados en el nombre del archivo ASPX tales como MiPagina.aspx.cs o MiPagina.aspx.vb (esta práctica se realiza automáticamente en Microsoft Visual Studio y otros entornos de desarrollo). Cuando se usa este estilo de programación, el desarrollador escribe el código correspondiente a diferentes eventos, como la carga de la página, o el clic en un control, en vez de un recorrido lineal a través del documento.

El modelo code-behind de ASP.NET marca la separación del ASP clásico y alienta a los desarrolladores a construir aplicaciones con la idea de presentación y contenido separados en mente. En teoría, esto permite a un diseñador web, por ejemplo, enfocarse en la creación del diseño con menos posibilidades de alterar el código de programación mientras lo hace. Esto es similar a la separación en el Modelo Vista Controlador.

6.3.2.3 Controles de usuario.

ASP.NET permite la creación de componentes reutilizables a través de la creación de Controles de Usuario (User Controls). Un control de usuario sigue la misma estructura que un formulario web, excepto que los controles derivan de la clase System.Web.UI.UserControl, y son almacenados en archivos ASCX. Como los archivos ASPX, un ASCX contiene etiquetas HTML o XHTML, además de etiquetas para definir controles web y otros controles de usuario. También pueden usar el modelo code-behind.

Los programadores pueden agregar sus propias propiedades y métodos, y manejadores de eventos. Un mecanismo de eventos en burbuja proporciona la capacidad de pasar un evento disparado por un control de usuario a la página que lo contiene.

6.3.2.4 Administración del estado.

Las aplicaciones ASP.NET son alojadas en un servidor web y se tiene acceso a ellas mediante el protocolo sin estado HTTP, que no guarda ninguna información sobre conexiones anteriores. Por lo tanto, si la aplicación requiere interacción entre conexiones, tiene que implementar su propia administración del estado. ASP.NET proporciona varias maneras de administrar el estado de las aplicaciones ASP.NET.

6.3.2.4.1 Estado de la aplicación.

El estado de la aplicación (Application state) es una colección de variables definidas por el usuario que son compartidas por todas las invocaciones de una aplicación ASP.NET. Estas son establecidas e inicializadas cuando el evento Application_OnStart se dispara en la carga de la primera instancia de las aplicaciones y están disponible hasta que la última instancia termina. Las variables de estado o variables de sesión de la aplicación son identificadas por nombres.

6.3.2.4.2 Estado de la sesión

El estado de la sesión (Session state) es una colección de variables definidas por el usuario, las cuales persisten durante la sesión de un usuario. Estas variables son únicas para diferentes instancias de una sesión de usuario, y son accedidas usando la colección Session. Las variables de sesión pueden ser preparadas para ser automáticamente destruidas después de un determinado tiempo de inactividad, incluso si la sesión no ha terminado. Del lado del cliente, una sesión de usuario es identificada por una cookie o codificando el ID de la sesión en la misma URL.

ASP.NET proporciona tres modos de persistencia para variables de sesión:

InProc

Las variables de sesión son mantenidas dentro del proceso. Sin embargo, en este modo, las variables son destruidas cuando el proceso ASP.NET es reciclado o terminado.

StateServer

En este modo, ASP.NET ejecuta un servicio de Windows separado que mantiene las variables de estado. Como esta administración de estado ocurre fuera del proceso ASP.NET, tiene un impacto negativo en el rendimiento, pero permite a múltiples instancias de ASP.NET compartir el mismo estado del servidor, permitiendo que una aplicación ASP.NET pueda tener su carga balanceada y escalada en múltiples servidores. También, como el servicio de administración del estado se ejecuta independiente de ASP.NET, las variables pueden persistir a través de las finalizaciones del proceso ASP.NET.

SQL Server

En este modo, las variables de estado son almacenadas en un servidor de base de datos, accesible usando SQL. Las variables de sesión pueden persistir a través de finalizaciones de procesos también en este modo.

6.3.2.4.3 Estado de la vista

El estado de la vista (View state) se refiere al mecanismo de administración de estado a nivel de página, que es utilizado por las páginas HTML generadas por las aplicaciones ASP.NET para mantener el estado de los controles de los formularios web y los widgets. El estado de los controles es codificado y mandado al servidor en cada envío del formulario en un campo oculto conocido como VIEWSTATE. El servidor envía de regreso las variables para que cuando la página sea renderizada de nuevo, los controles volverán a su último estado. Del lado del servidor, la aplicación puede cambiar el estado de la vista, si los resultados del procesamiento actualizan el

estado de cualquier control. Los estados de los controles individuales son decodificados en el servidor, y están disponibles para su uso en ASP.NET usando la colección ViewState.

6.3.2.5 Motor de plantillas.

Al ser liberado, ASP.NET carecía de un motor de plantillas. Debido a que .NET framework es orientado a objetos y permite herencia, muchos desarrolladores podrían definir una nueva clase que herede desde "System.Web.UI.Page", escribir métodos en ella que generen HTML y hacer que las páginas de su aplicación hereden de esta clase. Aunque esto permite que haya elementos comunes dentro de un sitio web, añade complejidad y mezcla código fuente con lenguaje de etiquetas. Además, este método puede visualizarse únicamente al ejecutar la aplicación, no mientras se está diseñando. Otros desarrolladores usan la inclusión de archivos y otros trucos para evitar la implementación de enlaces de navegación y otros elementos en cada página.

Las páginas hijas también usan esos controles ContentPlaceHolder, que deben ser relacionados con el ContentPlaceHolder de la página maestra que contiene a esta página hija. El resto de la página está definido por las partes compartidas de la página maestra. Todo el lenguaje de marcado y controles de servidor en la página de contenido deben ser colocadas dentro del control ContentPlaceHolder.

Cuando una solicitud es hecha por una página de contenido, ASP.NET mezcla la salida de la página de contenido con la salida de la página maestra, y envía el resultado al usuario.

La página maestra permanece completamente accesible a la página del contenido. Esto significa que la página de contenidos puede manipular los encabezados, cambiar el título, configurar la cache, etc. Si la página maestra expone propiedades públicas o métodos, el contenido de la página puede utilizar estos también.

Hoy en día con el nuevo producto de Microsoft WebMatrix, se dio a conocer el nuevo motor de plantillas conocido como Razor. Ya antes el IDE de Visual Studio incluía otros motores. Facilitando a las páginas ASP.NET Modelo Vista Controlador.

6.3.2.5.1 Otros archivos.

Otras extensiones de archivo asociadas con las diferentes versiones de ASP.NET incluyen:

Extensión	Versión requerida	Descripción
asax	1.0	Global.asax, usada para la lógica a nivel de aplicación[15]
ascx	1.0	Controles de usuario web: Controles personalizados para ser colocados en páginas web
ashx	1.0	Manejadores HTTP personalizados
asmx	1.0	Páginas de servicios web
axd	1.0	Cuando está habilitado en el web.config la solicitud de trace.axd genera trazas de salida a nivel de aplicación. También es usado para el manejador especial webresource.axd que permite a los desarrolladores de controles/componentes empacar un control/componente con imágenes, script, CSS, etc. para el desarrollo de un archivo único (un 'ensamblado')
browser	2.0	Archivos de capacidades del navegador almacenadas en formato XML; introducido en la versión 2.0. ASP.NET incluye muchos de estos por defecto, para admitir a los navegadores web comunes. Estos especifican que navegadores tienen que capacidades, así que ASP.NET 2 puede automáticamente personalizar y optimizar su salida de acuerdo al navegador. Los archivos especiales .browser están disponibles en descarga libre para manejar, por ejemplo, el validador de la W3C. Reemplaza la sección BrowserCaps que se encontraba en el archivo machine.config en ASP.NET 1.x
config	1.0	web.config es el único archivo en una aplicación web especifica que usa esta extensión por defecto (machine.config tiene efectos similares en un servidor web y todas las aplicaciones en el), sin embargo ASP.NET proporciona la facilidad de crear y utilizar otros archivos config. Son almacenados en formato XML
cs/vb	1.0	Archivos de código fuente (cs indica C#, vb indica Visual Basic). Los archivos code-behind predominantemente tienen la extensión ".aspx.cs" o ".aspx.vb" para los dos lenguajes más comunes. Otros archivos de código (que frecuentemente contienen bibliotecas de clases) pueden también existir en las carpetas web con las extensiones cs/vb. En ASP.NET 2 estos deberían estar colocados dentro de la carpeta App_Code donde son dinámicamente compilados y están disponibles para toda la aplicación
dbml	3.5	Archivo de clases de datos LINQ a SQL
master	2.0	Archivo de página maestra
resx	1.0	Archivos de recursos para localización y globalización. Los archivos de recursos pueden ser globales (por ejemplo, mensajes) o locales, que están hechos específicamente para un solo archivo aspx o ascx.
sitemap	2.0	Archivos de configuración sitemap
skin	2.0	Archivos de temas
svc	3.0	Archivos de servicio de Windows Communication Foundation

6.3.2.6 Estructura de directorios

En general, la estructura de directorios de ASP.NET puede ser determinada por las preferencias del desarrollador. Aparte de unos pocos nombres de directorios reservados, el sitio puede expandirse a cualquier número de directorios. La estructura es típicamente reflejada directamente en las urls.

Los nombres de directorios especiales (a partir de ASP.NET 2.0 son):

App_Browsers

Contiene archivos de definición específicos para navegadores.

App_Code

Es un directorio para códigos. El servidor ASP.NET automáticamente compilará los archivos (y subdirectorios) en esta carpeta en un ensamblado que es accesible desde cualquier página del sitio. App_Code es típicamente usada para código de acceso a datos, código de modelo o código de negocios. También cualquier manejador http específico para el sitio e implementación de módulos y servicios web van este directorio. Como alternativa a utilizar App_Code el desarrollador puede optar por proporcionar un ensamblado independiente con código precompilado.

App_Data

Directorio por defecto para las bases de datos, tales como archivos mdb de Microsoft Access y archivos mdf de Microsoft SQL Server. Este directorio es usualmente el único con permisos de escritura en la aplicación.

App_LocalResources

Contiene archivos de recursos localizados para páginas individuales del sitio.

App_GlobalResources

Contiene archivos resx con recursos localizados disponibles para cada página del sitio. Este es donde el desarrollador ASP.NET típicamente almacenara mensajes que serán usados en más de una página.

App_Themes

Usado para temas alternativos del sitio.

App_WebReferences

Usado para archivos de descubrimiento y archivos WSDL para referencias a servicios web para ser consumidos en el sitio.

Bin

Contiene código compilado (archivos .dll) para controles, componentes, y otro código que pueda ser referenciado por la aplicación. Cualquier clase representada por código en la carpeta Bin es automáticamente referenciada en la aplicación. Son archivos o librerías que tienen como principal acción ejecutar una función cuando estas son llamadas o se invocan.

6.3.3 Extensiones

6.3.3.1 ASP.NET AJAX.

ASP.NET AJAX, anteriormente llamado Atlas, es un conjunto de extensiones para ASP.NET desarrollado por Microsoft para implementar la funcionalidad de Ajax.

ASP.NET AJAX fue liberado en enero de 2007 después de un largo periodo de pruebas. Fue subsecuentemente incluido con la versión 3.5 del .NET Framework, que fue liberada junto con Visual Studio 2008 en noviembre de 2007.

6.3.3.2 ASP.NET MVC Framework

Es una extensión de páginas ASP.NET utilizando la arquitectura MVC.

6.3.3.3 ASP.NET Razor.

También conocido como ASP.NET Web Pages, es una alternativa a Web Forms diseñada para ser utilizada con MVC.

6.3.3.4 ASP.NET Dynamic Data

Es un Framework inspirado en Ruby on Rails para la plataforma de Microsoft, que viene como una extensión de ASP.NET, que se puede utilizar para construir aplicaciones web orientadas a los datos. Expone las tablas de una base de datos mediante la codificación en la URI del servicio web ASP.NET, y los datos de la tabla se representan automáticamente a HTML. El proceso de renderizado se puede controlar con el uso de plantillas de diseño personalizadas. Internamente, se descubre el esquema de base de datos mediante el uso de los metadatos de la base de datos.

6.3.3.5 ASP.NET Web API

Es un API HTTP para exposición de servicios Web RESTful.

6.3.3.6 ASP.NET SignaIR

Es una nueva biblioteca para desarrolladores ASP.NET que hace el desarrollo Web en tiempo real funcionalmente fácil. Permite una comunicación bidireccional entre el cliente y el servidor. Los servidores ahora pueden insertar el contenido a los clientes conectados al instante cuando estén disponible. Soporta Web Sockets. Incluye APIs para la gestión de la conexión (por ejemplo, conectar y desconectar eventos), agrupación de conexiones y autorización.

6.3.4 Modelos de programación en ASP.NET

Actualmente, ASP.NET soporta tres modelos de programación: ASP.NET Web Forms, ASP.NET MVC y ASP.NET Web Pages. Aunque los tres modelos de programación se ejecutan sobre la misma base de ASP.NET, cada uno de ellos estructura la aplicación de maneras completamente distintas, promueve metodologías de desarrollo diferentes y se adapta a perfiles de desarrolladores distintos. Algunas características que son virtudes en unos modelos de programación, pueden ser consideradas debilidades en el otro. ¿Qué es más importante, desarrollar a un gran nivel de abstracción o tener control total cada uno de los aspectos de la

aplicación? Simplicidad vs. Control. Flexibilidad vs. Eficiencia. Estas son las compensaciones que hay que baremar a la hora de elegir. En esta serie de artículos repasaremos las diferencias entre los tres modelos de programación, y los escenarios favorables a cada uno de ellos.

Es importante recalcar que el hecho de elegir uno de los modelos de programación al comenzar un proyecto de ASP.NET no excluye necesariamente a los otros, sino que es posible tener aplicaciones "híbridas" y en muchos casos tendrá todo el sentido desarrollar ciertas partes de la aplicación con un modelo de programación y otras partes con otro modelo distinto.

ASP.NET Web Forms fue el primero de los tres modelos de programación en existir, y proporciona un gran nivel de abstracción con un modelo de programación familiar basado en eventos y controles que favorece la productividad mediante la programación declarativa reduciendo la cantidad de código necesaria para implementar una determinada funcionalidad.

ASP.NET MVC se concibió como alternativa a Web Forms y proporciona un modelo de programación basado en el popular patrón de arquitectura MVC. Entre sus principales características destacan su completa integración con pruebas unitarias y su separación más clara entre la lógica de presentación, la lógica de negocio y la lógica de acceso a datos.

ASP.NET Web Pages es el más reciente de los tres modelos de programación, y fue creado como respuesta a una creciente demanda de desarrolladores web sin experiencia previa con ASP.NET, cuya iniciación en ASP.NET Web Forms o MVC les suponía una inversión inicial de tiempo demasiado grande. Web Pages proporciona un modelo de programación más simple y rápida de aprender, sin renunciar a toda la funcionalidad y flexibilidad de ASP.NET.

6.3.5 Uso actual del lenguaje

En la actualidad una aplicación .NET puede ejecutarse de dos formas distintas: Aplicaciones cliente/servidor: Estas aplicaciones están típicamente en formato de ejecutables

compilados. Estos pueden integrar toda la riqueza de una interfaz de usuario, tal es el caso de las aplicaciones de desempeño y productividad, pero no se reúne la lógica de negocio como un recurso que se pueda reutilizar. Además acostumbran ser menos gestionables y escalables que las demás aplicaciones.

Aplicaciones que utilizan el navegador: Dichas aplicaciones están caracterizadas por contar con una interfaz de web rica y muy útil. La interfaz gráfica integra varias tecnologías, las cuales son el HTML, XHTML, scripting, etc.; siempre y cuando el navegador que se esté utilizando soporte estas tecnologías.

6.4 Bootstrap

Bootstrap es un framework o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales. Tiene un soporte relativamente incompleto para HTML 5 y CSS3, pero es compatible con la mayoría de los navegadores web. La información básica de compatibilidad de sitios web o aplicaciones está disponible para todos los dispositivos y navegadores. Existe un concepto de compatibilidad parcial que hace disponible la información básica de un sitio web para todos los dispositivos y navegadores. Desde la versión 2.0 soporta diseños sensibles, es decir que el diseño gráfico de la página se ajusta dinámicamente, tomando en cuenta las características del dispositivo usado (Computadoras, tabletas, teléfonos móviles).

6.5 JavaScript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web

dinámicas. Sin embargo, existe una forma de JavaScript del lado del servidor (Serverside JavaScript o SSJS), que se usa en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets).

6.5.1 Características

- **Imperativo y estructurado*:*** es compatible con gran parte de la estructura de programación de C (por ejemplo, sentencias if, bucles for, sentencias switch, etc.). Con una salvedad, en parte: en C, el ámbito de las variables alcanza al bloque en el cual fueron definidas; sin embargo, JavaScript no es compatible con esto, puesto que el ámbito de las variables es el de la función en la cual fueron declaradas. Esto cambia con la versión de ECMAScript 2015, ya que añade compatibilidad con block scoping por medio de la palabra clave let. Como en C, JavaScript hace distinción entre expresiones y sentencias. Una diferencia sintáctica con respecto a C es la inserción automática de punto y coma, es decir, en JavaScript los puntos y coma que finalizan una sentencia pueden ser omitidos.

- **Dinámico:**

- **Tipado dinámico:** como en la mayoría de lenguajes de scripting, el tipo está asociado al valor, no a la variable. Por ejemplo, una variable x en un momento dado puede estar ligada a un número y más adelante, religada a una cadena. JavaScript es compatible con varias formas de comprobar el tipo de un objeto, incluyendo duck typing. Una forma de saberlo es por medio de la palabra clave typeof.

- **Objetual:** está formado casi en su totalidad por objetos. Los objetos son arrays asociativos, mejorados con la inclusión de prototipos. Los nombres de las propiedades de los objetos son claves de tipo cadena: obj.x = 10 y obj['x'] = 10 son equivalentes. Las propiedades y sus valores pueden ser creados, cambiados o eliminados en tiempo de ejecución. La mayoría de propiedades de un objeto (y aquellas que son incluidas

por la cadena de la herencia prototípica) pueden ser enumeradas a por medio de la instrucción de bucle for... in.

- **Evaluación en tiempo de ejecución:** incluye la función eval que permite evaluar expresiones expresadas como cadenas en tiempo de ejecución. Por ello se recomienda que eval sea utilizado con precaución y que se opte por utilizar la función JSON.parse() en la medida de lo posible, pues resulta mucho más segura.

- **Funcional:** las funciones se les suele llamar ciudadanos de primera clase; son objetos en sí mismos. Como tal, poseen propiedades y métodos, como: .call() y .bind(). Una función anidada es una función definida dentro de otra. Esta es creada cada vez que la función externa es invocada. Además, cada función creada forma una clausura; es el resultado de evaluar un ámbito conteniendo en una o más variables dependientes de otro ámbito externo, incluyendo constantes, variables locales y argumentos de la función externa llamante. El resultado de la evaluación de dicha clausura forma parte del estado interno de cada objeto función, incluso después de que la función exterior concluya su evaluación.

❖ **Prototípico**

- **Prototipos:** usa prototipos en vez de clases para el uso de herencia. Es posible llegar a emular muchas de las características que proporcionan las clases en lenguajes orientados a objetos tradicionales por medio de prototipos en JavaScript.

- **Funciones como constructores de objetos:** las funciones también se comportan como constructores. Prefijar una llamada a la función con la palabra clave new crear una nueva instancia de un prototipo, que heredan propiedades y métodos del constructor (incluidas las propiedades del prototipo de Object). ECMAScript 5 ofrece el método Object.create, permitiendo la creación explícita de una instancia sin tener que heredar automáticamente del prototipo de Object. La propiedad prototype del constructor determina el objeto usado para el prototipo interno de los nuevos objetos creados. Se

pueden añadir nuevos métodos modificando el prototipo del objeto usado como constructor.

6.6 Hojas de estilo en cascada (CSS)

Es una tecnología que permite crear páginas web de una manera más exacta. Gracias a las CSS se pueden incluir en las páginas web márgenes, tipos de letra, fondos, colores, etc. Las Hojas de Estilo en Cascada se escriben dentro del código HTML de la página web, solo en casos avanzados se pueden escribir en un archivo a parte y enlazar la página con ese archivo, pero igual se puede utilizar en un atributo en casi todas las etiquetas de HTML. El lenguaje CSS permite presentar, de manera estructurada, un documento que fue escrito en un lenguaje de marcado. Se usa especialmente en el diseño visual de un sitio web cuando las páginas se hallan escritas en XML o HTML. El CSS se desarrolló en distintos niveles. El CCS1 ya no se emplea, mientras que el CSS2 funciona como recomendación. El CSS3, que se divide en varios módulos, es el lenguaje que se está tomando como estándar. Lo que hace el CSS es encargarse de la descripción de las formas y de la sintaxis del lenguaje de marcado. De esta manera describe cómo se tienen que renderizar (generar las imágenes) los elementos que aparecen en pantalla.

6.7 DevExpress

Developer Express Inc. (DevExpress) es una compañía de desarrollo de software fundada en 1,998 con sede en Glendale, California .DevExpress comenzó inicialmente a producir controles de UI para Borland Delphi / C ++ Builder y controles ActiveX para Microsoft Visual Studio. Actualmente, DevExpress tiene productos dirigidos a desarrolladores que usan Delphi / C ++ Builder, Visual Studio y HTML 5 / JavaScript.

DevExpress controles de Windows Forms comparten muchas características comunes. Ellos utilizan los mismos métodos para controlar la apariencia configuraciones, datos, enfoques vinculantes mismos valores de formato y localización de recursos, etc.

6.7.1 XtraChart

XtraCharts suites un conjunto de controles de gráficos profesionales para. NET (para las formas tanto en plataformas Windows como ASP.NET, así como para XtraReports). Meticulosamente diseñado con una impresionante variedad de tipos de gráficos diferentes, un conjunto de indicadores específicos listo visuales, monitoreo en tiempo real, análisis comparativo / contrastivo, mientras que también es capaz de gráficos financieros, XtraCharts es la solución a todas sus necesidades de creación de gráficos.

Independientemente de su tipo de origen de datos (ya sea una base de datos, colección o red de pivote) o la cantidad y el tipo de datos de entrada, XtraCharts es la mejor herramienta para la visualización de todos los datos con capacidades de personalización casi infinitas. Con funciones inteligentes de automatización de alto nivel, XtraCharts hace de gráficos 2D y 3D más fácil que nunca, mientras que permanece flexible y eficiente mediante la manifestación de una interfaz amigable e intuitiva.

6.7.1.1 Características generales

Soporte para múltiples plataformas de destino: XtraCharts Suite incluye el ChartControl (para Windows Forms) y WebChartControl (para ASP.NET). Ambos ofrecen una funcionalidad similar, mientras que ofrece todas las ventajas de sus plataformas específicas.

Profundo apoyo para los productos DevExpress otros: PivotGridControl (ASPxPivotGrid) y XtraReports.

Soporte para múltiples tipos de gráficos diferentes: 2D y 3D tipos de gráficos desde básico (por ejemplo, barras y columnas, Punto y línea, Pie y re-direccionamiento, o Radar y Polar) a lo específico (por ejemplo, Burbuja, Gantt o Parcela Swift que es especialmente diseñado para su procesamiento inmediato de cantidades muy grandes de puntos de datos).

6.7.1.2 Enlace de Datos

- XtraCharts soporta una variedad de tipos de orígenes de datos comunes, que también incluye soporte completo + ADO.
- Modo independiente - Una serie puede ser rellenada con datos manualmente por especifican valores para los (X) e (Y) las coordenadas de los puntos. También es posible combinar unido y sin unir puntos de vista de la serie en un solo diagrama.
- Series Encuadernación Individual - XtraCharts puede mostrar datos de múltiples fuentes de datos, ya que cada serie se puede enlazar a su propia tabla de datos.
- Uso de las plantillas de la Serie - En este modo el ChartControl crea Visto serie automáticamente, y no hay necesidad de rellenar la colección Serie manualmente.
- Serie de datos de filtrado - Esta característica le permite aplicar criterios de filtro para los datos mostrados por series específicas, lo que puede evitar la inundación del control de gráfico con datos irrelevantes.
- Datos de la Serie Clasificación - Puede ordenar los datos dentro de una serie, ya sea por sus argumentos o valores.
- Serie de datos de resumen - Puede aplicar diferentes funciones de resumen para los datos de una serie.

Impresión y Exportación - Si usted es dueño de la Biblioteca XtraPrinting , usted puede exportar el gráfico a: un archivo de imagen (múltiples formatos de archivo son compatibles), un archivo PDF; una imagen incrustada en una página HTML, una imagen incrustada en una hoja de cálculo de Microsoft Excel, una página impresa.

6.8 Microsoft SQL Server

Microsoft SQL Server es un conjunto completo de tecnologías y herramientas específicas para la empresa que permiten que el personal obtenga de la información el máximo valor al menor costo de propiedad (MICROSOFT, 2013).

SQL Server es un sistema de manejo de bases de datos del modelo relacional, desarrollado por Microsoft. El lenguaje de desarrollo utilizado es Transact-SQL (TSQL) el cual puede ser utilizado por línea de comandos o mediante la interfaz gráfica de Management Studio, este es una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).

El motor de base de datos de SQL Server ofrece almacenamiento más seguro y confiable tanto para los datos relacionales como estructurados, lo que permite crear y administrar aplicaciones de datos altamente disponibles y con mayor rendimiento. Además, combina lo mejor en análisis, información, integración y notificación.

6.8.1 Características de Microsoft SQL Server

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Incluye un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Permite administrar información de otros servidores de datos.

6.8.2 Herramientas de SQL Server

Aparte de sus capacidades elementales como herramienta de gestión de bases de datos relacionales (crear tablas, definir las relaciones, gestión de transacciones, crear índices etc.), posee otras herramientas las cuales se enlistan a continuación:

- **Respaldos y recuperaciones:** Existe una herramienta integrada en el SQL Server que posibilita un respaldo completo o diferencial, de acuerdo con el modelo de recuperación (Recovery Model) predefinido a la base de datos; y una recuperación completa o a un punto de tiempo. Aparte de un respaldo de la base de datos se puede respaldarlos a través de un guion (con o sin los datos). A partir de 2008, se puede comprimir los archivos de respaldo.

- **Compresión:** A partir de 2008 se añadió la opción de compresión que posibilita comprimir el tamaño físico de las tablas y los índices, y utilizar más eficientemente el volumen de los discos rígidos y reducir operaciones de IO (lo que aumenta la carga del CPU). Como ya ha sido mencionado, se puede comprimir también los archivos de respaldo.
- **Replicación, alta disponibilidad, y recuperación de desastres:** Algunas herramientas posibilitan crear réplicas parciales o completos de las bases de datos, mejorar la disponibilidad, y recuperar de desastres; aparte de la opción de respaldar y recuperar las bases de datos, una opción que se supone que es muy lenta; se mencionan a continuación:
  - **La replicación:** que posibilita crear una réplica sincronizada de la base de datos
  - **El Mirroring***:* que ejecuta en el servidor de espejo cada actualización que se ejecuta en el servidor de origen
  - **Log Shipping:** que posibilitan almacenar una copia sincronizada a través de archivos de registro (Log) con todas las actualizaciones en el servidor de origen.
  - Herramientas de alta disponibilidad y recuperación de desastres (HADR, desde 2012) que solucionan problemas de Mirroring en relación con el tiempo de reacción de los fallos técnicos y la disponibilidad de las copias en tiempos de paz.

- **El agente y la programación de tareas:** El agente es el servicio encargado de la programación de tareas, y se encarga de ejecutarlas independientemente. Generalmente ejecuta tareas de mantenimiento, tareas complejas de ETL, respaldos, etc.
- **Service Broker:** Una tecnología que fue agregada en 2005, la cual implementa arquitectura orientada a servicios, y que posibilita ejecuciones asíncronas: primero para enviar mensajes entre distintas aplicaciones que se ejecutan simultáneamente, pero también para ejecutar procedimientos asíncronamente, en la manera de dispar y olvidar: un procedimiento que se ejecuta en una sesión diferente de la sesión que lo inicio, y ambos procedimientos siguen ejecutado independientemente uno del otro.
- **Enviar Correos Electrónicos:** SQL Server tiene la capacidad de enviar correos electrónicos a través del código. Esta capacidad se utiliza generalmente para enviar alertas sobre problemas en el sistema, pero también cuando el proceso termino exitosamente.
- **Full Text Search (Búsqueda de Textos completos):** Una herramienta que posibilita indexar columnas textuales como textos y no solo como cadenas; y ejecutar búsquedas complejas dependientes en el sentido del texto y en el idioma.
- **Rastrear:** Estas herramientas incluyen el Trace que posibilita rastrear actividades con el fin de mantener cargas y fallos, y seguridad de datos. El Profiler que posibilita rastrear los comandos que se ejecutan y los eventos que se ocurren en el servidor, y el Extended Events que fue agregado en 2008 y cambia el profiler gracias a su baja consumición de recursos y la influencia sobre el rendimiento del servidor.
- **Herramientas de Inteligencia empresarial:** Una instalación típica incluye también las herramientas de BI (Inteligencia empresarial):

  - **SSIS (SQL Server Integration Services):** Una herramienta de ETL que posibilita la extracción de datos de distintos orígenes (no solo SQL Server), la transformación de dichos datos, y la carga (generalmente pero no obligatoriamente almacén de datos).
  - **SSAS (SQL Server Analysis Services):** Una herramienta para crear Bases de Datos Multidimensionales, que se puede explorar mediante extracciones de datos

en distintos niveles de agrupación, profundización (Drill Down) de una suma a sus detalles, y utilización de MDX (un lenguaje parecido a SQL, adaptado a bases de datos multidimensionales).

➢ **SSRS (SQL Server Reporting Services):** Una herramienta para crear y dar formato a informes, otorgar derechos de contemplación en ellos, y su distribución. Se puede contemplarlos con un Navegador web, y se puede exportarlos a archivos de Excel, PDF, etc. los datos se extraen generalmente del almacén de datos o del OLAP.

En nuestro proyecto hemos llegado a la necesidad de usar el modelo de base de datos relacional ya que es el que más se acopla a nuestras necesidades, no obstante, decidimos mencionar los diferentes modelos de base de datos para dar un mayor conocimiento en el concepto de las BBDD.

6.8.3 Modelos de bases de datos

Un **modelo de base de datos** o esquema de base de datos es la estructura o el formato de una base de datos, descrita en un lenguaje formal soportada por el sistema de gestión de bases de datos. En otras palabras, un "modelo de base de datos" es la aplicación de un modelo de datos usado en conjunción con un sistema de gestión de bases de datos. Los esquemas generalmente son almacenados en un diccionario de datos. Aunque un esquema se defina en un lenguaje de base de datos de texto, el término a menudo es usado para referirse a una representación gráfica de la estructura de la base de datos.

Base de Datos Jerárquica: Puede representar dos tipos de relaciones entre los datos: relaciones de uno a uno y relaciones de uno a muchos. Este modelo tiene forma de árbol invertido en el que una rama puede tener varios hijos, pero cada hijo sólo puede tener un padre.

Base de Datos De red: Este modelo permite la representación de muchos a muchos, de tal forma que cualquier registro dentro de la base de datos puede tener varias ocurrencias superiores a él. El modelo de red evita redundancia en la información, a través de la incorporación de un tipo

de registro denominado el conector. En el modelo en red se representa el mundo real mediante registros lógicos que representan a una entidad y que se relacionan entre sí por medio de flechas.

Base de Datos Transaccionales: Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

Bases de datos relacionales: Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

Las bases de datos relacionales pasan por un proceso al que se le conoce como normalización de una base de datos, el cual es entendido como el proceso necesario para que una base de datos sea utilizada de manera óptima. Entre las ventajas de este modelo están:

- Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.

- Garantiza la integridad referencial: Así al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización por ser más comprensible y aplicable. Para llevar a cabo nuestro proceso de creación y manipulación de una base de datos hemos decidido utilizar el gestor "MICROSOFT SQL SERVER 2014", ya que nos proporciona las herramientas necesarias para cumplir con las expectativas que nos hemos planteado para con nuestro trabajo de investigación. Debido a que el Sistema cuenta con un sistema de usuario, es de gran importancia proteger las contraseñas, con el fin de prevenir posibles vulnerabilidades en el servidor por parte de terceros, de esta manera si alguien no autorizado accede a dicho servidor no podrá ver las contraseñas sino su encriptación.

  Para esto hay un sin número de algoritmos de encriptación, como lo es AES, MD5, a continuación se da una breve explicación. (CEBALLOS, 2007)

6.9 Encriptar

¿Qué es encriptar?

Hoy en día, el concepto se entiende como tomar información y hacerla ilegible a los demás. También es conocido como cifrado. Puede ser algo simple como texto, aunque la evolución de la comunicación en línea ahora implica el uso de imágenes, videos, sonidos y más.

6.9.1 AES

¿Cómo funciona el AES?

El estándar de cifrado avanzado (AES, por sus siglas en inglés) describe una fórmula matemática o algoritmo, para la conversión de datos electrónicos en una forma ininteligible, denominada texto cifrado. El texto cifrado no puede ser leído por cualquier persona que no sea el destinatario.

El AES funciona alimentando una clave de cifrado, esencialmente una cadena de dígitos en el algoritmo de cifrado y realizando de una serie de operaciones matemáticas basadas en esa clave de cifrado.

6.9.1.1 Operación básica

El AES puede ser descrito como un bloque cifrado iterativo y simétrico. El AES utiliza una estructura de bucle para realizar repetidamente reordenamientos de datos, o permutaciones. El bucle reemplaza una unidad de datos con otra para datos de entrada. La rutina de cifrado utiliza la misma clave para cifrar y descifrar los datos, y aplica esa clave a los bloques de datos de longitud.

6.9.1.2 Programa clave

La rutina de cifrado de AES almacena la clave de cifrado principal en una matriz. Una matriz es un grupo de objetos con los mismos atributos que pueden ser abordados de forma individual. La matriz consta de cuatro las, conteniendo cada uno cuatro, seis u ocho bytes, dependiendo del tamaño de la clave. Tras bambalinas, la rutina de cifrado utiliza esta matriz para generar una tabla, conocida como un programa clave, que contiene varias claves. Estas claves se denominan claves redondas para distinguirlas de la llave maestra original.

6.9.1.3 Matriz de estado

El AES utiliza una clave de cifrado que puede ser 1 28, 1 92 o 256 bits de largo, y se aplica en unidades de datos, llamados bloques, cada uno de los cuales es de 1 28 bits de largo. El algoritmo AES comienza copiando cada bloque de 1 6 bits en una matriz bidimensional llamada el Estado, para crear una matriz de bytes de 4x4. El algoritmo realiza una operación exclusiva "O"

que devuelve "verdadero" si uno u otro de sus operandos es verdadero. Esto se conoce como "AddRoundKey", y está entre las primeras cuatro las del programa clave y la matriz de Estado.

6.9.1.4 Operaciones matemáticas

Tras la operación inicial exclusivo "O", el algoritmo de cifrado AES entra en su bucle principal, en el que realiza repetidamente cuatro operaciones matemáticas diferentes en la matriz de Estado: "SubBytes", "ShiftRows", "MixColumns" y "AddRoundKey". Estas operaciones emplean una combinación de suma, multiplicación, rotación y sustitución para cifrar cada byte en la matriz de Estado. El bucle principal se ejecuta 10, 12 o 14 veces dependiendo del tamaño de la clave de cifrado. Una vez que se completa la ejecución, el algoritmo copia la matriz de estado a su salida en forma de texto cifrado. (https://techlandia.com/funciona-aes-info_215975/)

6.9.2 MD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje

Es un algoritmo de reducción criptográfico de 128 bits ampliamente usado. En criptografía, MD5 (abreviatura de Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado. Uno de sus usos es el de comprobar que algún archivo no haya sido modificado.

6.9.2.1 Historia

Es uno de los algoritmos de reducción criptográficos diseñados por el profesor Ronald Rivest del MIT (*Massachusetts Institute of Technology*, Instituto Tecnológico de Massachusetts). Fue desarrollado en 1991 como reemplazo del algoritmo MD4 después de que Hans Dobbertin descubriese su debilidad. A pesar de su amplia difusión actual, la sucesión de problemas de seguridad detectados desde que, en 1996, Hans Dobbertin anunciase una colisión de *hash*, plantea una serie de dudas acerca de su uso futuro.

Codificación: La codificación del MD5 de 128 bits es representada típicamente como un número de 32 dígitos hexadecimal. El siguiente código de 28 bytes ASCII será tratado con MD5 y veremos su correspondiente *hash* de salida:

```
MD5("Esto sí es una prueba de MD5") = e99008846853ff3b725c27315e469fbc
```

Un simple cambio en el mensaje nos da un cambio total en la codificación *hash*, en este caso cambiamos dos letras, el «sí» por un «no».

```
MD5("Esto no es una prueba de MD5") = dd21d99a468f3bb52a136ef5beef5034
```

6.9.2.2 Aplicaciones

Los resúmenes MD5 se utilizan extensamente en el mundo del software para proporcionar la seguridad de que un archivo descargado de Internet no se ha alterado. Comparando una suma MD5 publicada con la suma de comprobación del archivo descargado, un usuario puede tener la confianza suficiente de que el archivo es igual que el publicado por los desarrolladores. Esto protege al usuario contra los 'Caballos de Troya' o 'Troyanos' y virus que algún otro usuario malicioso pudiera incluir en el software.

Al mismo tiempo el algoritmo MD5 se utiliza para calcular el *hash* de las claves de los usuarios. En el disco se guarda el resultado del MD5 de la clave que se introduce al dar de alta un usuario, y cuando éste quiere entrar en el sistema se compara el *hash* MD5 de la clave introducida con el *hash* que hay guardado en el disco duro. Si coinciden, es la misma clave y el usuario será autenticado.

El MD5 también se puede usar para comprobar que los correos electrónicos no han sido alterados usando claves públicas y privadas.

6.10 Visual Basic

Visual Basic (VB) es un lenguaje de programación dirigido por eventos, desarrollado por Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo.

La última versión fue la 6, liberada en 1998, para la que Microsoft extendió el soporte hasta marzo de 2008.

En 2001 Microsoft propuso abandonar el desarrollo basado en la API Win32 y pasar a un framework o marco común de librerías, independiente de la versión del sistema operativo .NET Framework, a través de Visual Basic .NET (y otros lenguajes como C Sharp (C#) de fácil transición de código entre ellos); fue el sucesor de Visual Basic 6.

Aunque Visual Basic es de propósito general, también provee facilidades para el desarrollo de aplicaciones de bases de datos usando Data Access Objects, Remote Data Objects o ActiveX Data Objects.

6.10.1 Historia

Todas las versiones de Visual Basic para Windows son muy conocidas, aunque la Microsoft Visual Basic 1.0 desarrollada para el sistema operativo MS-DOS (ediciones Profesional y Estándar), que data de 1992, fue menos difundida. Esta proveía un entorno que, aunque en modo texto, incluía un diseñador de formularios en el que se podían arrastrar y soltar distintos controles.

La última versión que únicamente generaba aplicaciones de 16 bits fue la 3.0 y no incluía una biblioteca detallada de componentes para toda clase de usos. Durante la transición de los sistemas Windows 3.11 a Windows 95, en 1995: hizo su aparición la versión 4.0 de Visual Basic; esta

podía generar programas tanto de 16 como de 32 bits, a partir del mismo código fuente, aunque a costa de un gran aumento en el tamaño de los archivos necesarios en tiempo de ejecución ("runtime"). Además, se sustituyeron los controles denominados VBX por los nuevos OCX. Con la siguiente versión, la 5.0, se estuvo a punto de implementar por primera vez la posibilidad de compilar a código nativo, obteniendo una mejora de rendimiento considerable. Tanto esa como la sucesora 6.0 soportaban ciertas características propias de los lenguajes orientados a objetos, pero carecían de algunas importantes, tales como herencia y sobrecarga; pero, de hecho, no fue pensado como lenguaje orientado a objetos. La versión 6.0, que puede generar código ejecutable directo en 32 bits, todavía continúa utilizándose masivamente, y es compatible con las últimas versiones de los sistemas Windows, como Windows 7 y Windows 8.

Visual Basic evolucionó para integrar la plataforma .NET; allí perdió su propia identidad como lenguaje único adquirible, pasando a integrar un paquete de productos, llamado precisamente Microsoft .NET; dentro de ese paquete o framework se encuentra el nuevo y llamado Visual Basic .NET, que trabaja sobre el entorno Microsoft Visual Studio. Esta nueva versión del lenguaje posee profundas diferencias en la forma de programar respecto de Visual Basic 6, pero gran semejanza en su sintaxis básica.

Cabe mencionar que, aunque fue menos conocido, se desarrolló también una versión gratuita de Visual Basic 5.0, orientada al desarrollo de controles y componentes; su nombre específico era Microsoft Visual Basic 5.0 Control Creation Edition (Visual Basic 5 CCE). También hubo versiones orientadas al desarrollo de aplicaciones para dispositivos móviles basados en Windows CE y Pocket PC, conocidas como Em bedded (Visual Basic).

6.10.2 Características

Los compiladores de Visual Basic generan código que requiere una o más librerías de enlace dinámico para que funcione, conocidas comúnmente como DLL (sigla en inglés de *Dynamic-Link Library*); en algunos casos reside en el archivo llamado MSVBVMxy.DLL (siglas de "MicroSoft Visual Basic Virtual Machine x.y", donde x.y es la versión) y en otros en VBRUNXXX.DLL

("Visual Basic Runtime X.XX"). Estas bibliotecas DLL proveen las funciones básicas implementadas en el lenguaje, conteniendo rutinas en código ejecutable que son cargadas *bajo demanda* en tiempo de ejecución. Además de las esenciales, existe un gran número de bibliotecas del tipo DLL con variedad de funciones, tales como las que facilitan el acceso a la mayoría de las funciones del sistema operativo o las que proveen medios para la integración con otras aplicaciones.

Dentro del mismo Entorno de desarrollo integrado (IDE) de Visual Basic se puede ejecutar el programa que esté desarrollándose, es decir en modo intérprete (en realidad pseudo-compila el programa muy rápidamente y luego lo ejecuta, simulando la función de un intérprete puro). Desde ese entorno también se puede generar el archivo en código ejecutable (exe); ese programa así generado en disco puede luego ser ejecutado sin requerir del ambiente de programación (incluso en modo stand alone), aunque sí será necesario que las librerías DLL requeridas por la aplicación desarrollada se encuentren también instaladas en el sistema para posibilitar su ejecución.

El propio Visual Basic provee soporte para empaquetado y distribución; es decir, permite generar un *módulo instalador* que contiene al programa ejecutable y las bibliotecas DLL necesarias para su ejecución. Con ese módulo la aplicación desarrollada se distribuye y puede ser instalada en cualquier equipo (que tenga un sistema operativo compatible).

Así como bibliotecas DLL, hay numerosas aplicaciones desarrolladas por terceros que permiten disponer de variadas y múltiples funciones, incluso mejoras para el propio Visual Basic; las hay también para el empaquetado y distribución, y hasta para otorgar mayor funcionalidad al entorno de programación (IDE).

6.10. 3 Entorno de desarrollo

Existe un único entorno de desarrollo para Visual Basic, desarrollado por Microsoft: Microsoft Visual Basic x.0, correspondientes a versiones desde la 2.0 hasta la 20.0, (con respectivas diferencias entre versiones del lenguaje).

El entorno de desarrollo es muy similar al de otros lenguajes. Realizando una instalación típica del producto, las características básicas se presentan de la siguiente forma:

En la parte superior aparecen tres elementos, en este orden: la *barra de título* donde figura el nombre del proyecto en curso y su estado (diseño o ejecución); la barra de menú con 1 3 opciones desplegables y una barra de herramientas; esta última se puede personalizar, posibilitando la inclusión de prácticamente la totalidad de los comandos del IDE.

En la parte central, cubriendo la franja de mayor área, se encuentra el *espacio de trabajo*. Este incluye y muestra las *ventanas del proyecto*, las vistas del código fuente de los módulos, los objetos y los controles que contienen las ventanas de la aplicación y el *panel de controles*.

El panel de controles, que aunque es móvil normalmente está ubicado a la derecha, por defecto cuenta con los siguientes controles:

PictureBox: Caja de imágenes
Label: Etiqueta
***TextBox*:** Caja de texto
Frame: Marco
CommandButton: Botón de comando
CheckBox: Casilla de verificación
OptionButton: Botón de opción
ComboBox: Lista desplegable
ListBox: Lista
HScrollBar: Barra de desplazamiento horizontal
VScrollBar: Barra de desplazamiento vertical
Timer: Temporizador
DriveListBox: Lista de unidades de disco
DirListBox: Lista de directorios
FileListBox: Lista de archivos
Shape: Figura

Line: Línea

Image: Imagen

Data: Conexión a origen de datos

OLE: Contenedor de documentos embebidos compatibles con *Object Linking and Embedding*.

Además de los listados, se pueden agregar todo tipo de controles de terceros, y hay una gran cantidad de ellos que se proveen con el propio Visual Basic 6.0. Los controles vienen embebidos dentro de archivos con extensión OCX.

6.10.4 Ventajas

- Posee una curva de aprendizaje muy rápida.
- Integra el diseño e implementación de formularios de Windows.
- Permite usar con facilidad la plataforma de los sistemas Windows, dado que tiene acceso prácticamente total a laAPI de Windows, incluidas librerías actuales.
- Es uno de los lenguajes de uso más extendido, por lo que resulta fácil encontrar información, documentación y fuentes para los proyectos.
- Fácilmente extensible mediante librerías DLL y componentes ActiveX de otros lenguajes.
- Posibilita añadir soporte para ejecución de scripts, VBScript o JScript, en las aplicaciones mediante *Microsoft Script Control*.
- Tiene acceso a la API multimedia de DirectX (versiones 7 y 8). También está disponible, de forma no oficial, un componente para trabajar con OpenGL 1 .1.
- Existe una versión, VBA, integrada en las aplicaciones de Microsoft Office, tanto Windows como Mac, que permite programar macros para extender y automatizar funcionalidades en documentos, hojas de cálculo y bases de datos (Access).
- Si bien permite desarrollar grandes y complejas aplicaciones, también provee un entorno adecuado para realizar pequeños prototipos rápidos.

6.10.5 Desventajas

Las críticas hechas en las ediciones de Visual Basic anteriores a VB.NET son variadas; se citan entre ellas:

- Problema de versionado asociado con varias librerías runtime DLL, conocido como DLL Hell.
- Soporte pobre para programación orientada a objetos.
- Incapacidad para crear aplicaciones multihilo, sin tener que recurrir a llamadas de la API de Windows.
- Dependencia de complejas y frágiles entradas de registro COM.

La capacidad de utilizar controles en un único formulario es muy limitada en comparación a otras herramientas.

6.11 Arquitectura Cliente-Servidor

La arquitectura cliente-servidor es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

Algunos ejemplos de aplicaciones computacionales que usen el modelo cliente-servidor son el Correo electrónico, un Servidor de impresión y la World Wide Web.

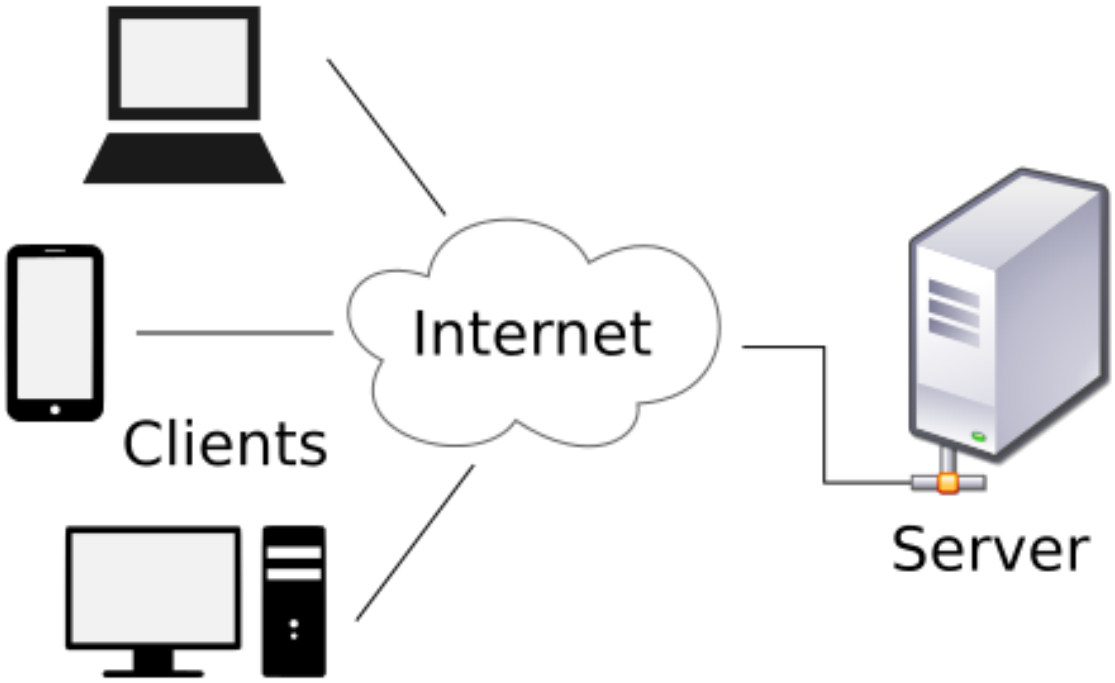
Características Principales:

En la arquitectura C/S el remitente de una solicitud es conocido como cliente. Sus características son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación (dispositivo maestro o amo).
- Espera y recibe las respuestas del servidor.

- Por lo general, puede conectarse a varios servidores a la vez.
- Normalmente interactúa directamente con los usuarios finales mediante una interfaz gráfica de usuario.

Esquema Cliente - Servidor



7. DISEÑO METODOLÓGICO

La ingeniería del software está centrada en obtener aplicaciones económicas y fiables que funcionen de manera más eficiente en máquinas reales. Nuestro sistema consta de diferentes herramientas que unidas entre sí permiten una mejor calidad en el procesamiento de los datos.

Por lo tanto, en este proyecto que lleva como tema ***Desarrollo de un sistema Web en ASP.net para la gestión de recursos humanos basada en nóminas en la empresa ACAPROF ubicada en Matagalpa Nicaragua durante el período Enero a Mayo de 2019***, hemos decidido trabajar en un lenguaje de programación orientado a la Web Visual Besas, desarrollado en la plataforma asp.net de Visual Studio 2013, para obtener una mejor seguridad y distribución de los datos, al mismo tiempo se utilizó como gestor de bases de datos: SQL Server 2014, para obtener de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para la empresa.

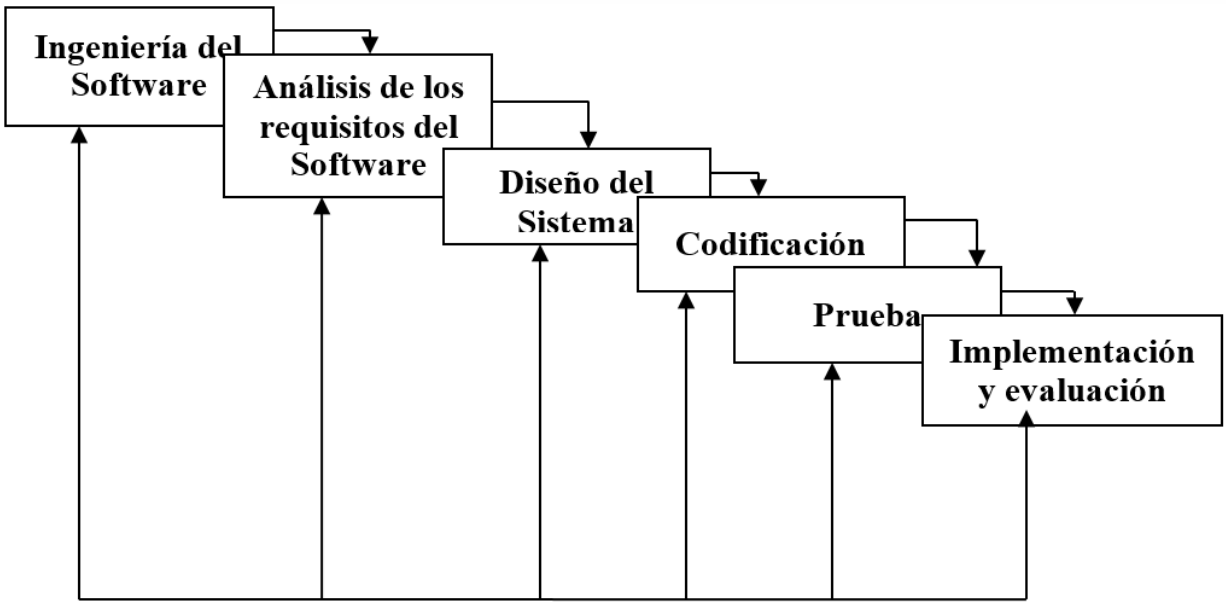
7.1 Metodología del Trabajo

El método de desarrollo que hemos seleccionado para la elaboración de la aplicación es el método ciclo de vida clásico o en cascada ya que resulta conveniente en la identificación de las actividades a seguir en la elaboración de la aplicación.

Las etapas a desarrollar durante el proceso investigativo son:

1. Investigación preliminar o ingeniería en sistemas
2. Análisis de requisitos del software
3. Diseño del sistema
4. Codificación
5. Prueba
6. Implementación y evaluación

Ciclo de vida clásico o en cascada



1. **Investigación preliminar o ingeniería del sistema:** en esta fase se realiza el estudio de factibilidad técnica, económica y operacional de la aplicación.
2. **Análisis de requisitos del software:** se debe comprender la funcionalidad que debe tener la aplicación, así como su rendimiento y la interfaz requerida.

En esta fase se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. De esta fase surge una memoria llamada ERS (documento de especificación de requisitos software), que contiene la especificación completa de lo que debe hacer el sistema sin entrar en detalles internos.

Es importante señalar que en esta etapa se debe consensuar todo lo que se requiere del sistema y será aquello lo que seguirá en las siguientes etapas, no pudiéndose requerir nuevos resultados a mitad del proceso de elaboración del software.

3. **Diseño del sistema:** se diseñan procedimientos precisos para especificar la interfaz que utilizará la aplicación.

Descompone y organiza el sistema en elementos que puedan elaborarse por separado, aprovechando las ventajas del desarrollo en equipo. Como resultado contiene la descripción de la estructura relacional global del sistema y la especificación de lo que debe hacer cada una de sus partes, así como la manera en que se combinan unas con otras.

Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario, así como también los análisis necesarios para saber que herramientas usar en la etapa de Codificación.

4. **Codificación:** el diseño se traduce de forma precisa a la computadora, utilizando un lenguaje de programación de alto nivel, en nuestro caso Visual Studio 2013. Se debe codificar los ejercicios resueltos y propuestos, además de generar la interfaz gráfica de cada uno de ellos.

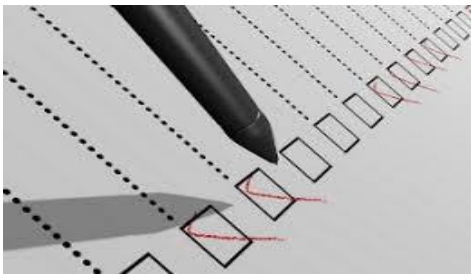
Es la fase en donde se implementa el código fuente, haciendo uso de prototipos, así como de pruebas y ensayos para corregir errores.

Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido.

5. **Prueba:** una vez generado el código respectivo para cada una de las guías, comienza la fase de prueba de cada una de ellas, confirmando que se puede ejecutar cada una de las prácticas propuestas.

Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente y que cumple con los requisitos, antes de ser entregado al usuario final.

6. **Implementación y evaluación:** una vez realizada la fase de prueba y después de haber hechos las correcciones en caso de ser necesario, se procede a la publicación de las guías.



Es la fase en donde el usuario final ejecuta el sistema, para ello el o los programadores ya realizaron exhaustivas pruebas para comprobar que el sistema no falle.

7. **Mantenimiento:** Debido a que el programa puede tener errores, puede no ser del completo agrado del cliente o puede necesitar, eventualmente acoplarse a los cambios en su entorno. Esto quiere decir que no se rehace el programa, sino que sobre la base de uno ya existente se realizan algunos cambios.

7.2 Recursos Disponibles y Necesarios

1. **Recursos Hardware**

Disponible: Para la elaboración de nuestro trabajo monográfico hemos utilizado una PC con las siguientes características:

- Procesador: Intel de 2 GHz
- Disco Duro: 500 GB
- Memoria Principal: 4 GB RAM

2. **Recursos Software**

Sistema Operativo: Windows versión 10,7

Entorno de desarrollo utilizado: Visual Studio 2013, Visual Basic

Otro tipo de Herramientas: DevExpres, JavaScript, Boostrap

Sistema Gestor de Base de Datos: Microsoft SQL Server.

8. ESPECIFICACIÓN DE REQUISITOS SOFTWARE

8.1 Especificación de requisitos software según el estándar de IEEE-8330

En este documento incorporaremos las Especificación de Requisitos Software (ERS) para el sistema Web en ASP.net para la gestión de recursos humanos basada en nóminas en la empresa ACAPROF ubicada en Matagalpa Nicaragua durante el período Enero a Mayo de 2019. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

8.1.1 Propósito

Este documento ha sido redactado con el fin de marcar las pautas generales y las especificaciones que deberá seguir el software a desarrollar, con el objetivo final de resolver las necesidades que el cliente ha planteado.

Servirá como canal de comunicación entre el Empleador y el empleado para llevar un sistema de nómina de la empresa ACAPROF.

- ✓ Ayudará a la administración de nómina de la forma más rápida posible
- ✓ Le dará una visión más profesional para la atención de los usuarios vía online.

8.1.2 Alcance del sistema

Se ha constatado que la necesidad de un sistema de nóminas. El objetivo de este software es gestionar, visualizar, organizar las nóminas, así como gestiones del personal como vacaciones, permisos, boletas de pagos, con la finalidad de optimizar el uso eficaz de su tiempo.

El sistema web será una aplicación que funcionará en un entorno WEB que permitirá administrar y consultar la información de los usuarios, planillas, vacaciones, descansos. Esta aplicación dará apoyo a los siguientes procesos:

- Autentificación de Usuario.
- Creación de Nuevas Empresas dentro del sistema.
- Creación de Departamento de área
- Creación de Puestos por departamento
- Creación de grupos de empresas
- Administración de empleados
- Administración de usuarios
- Administrar planillas
- Importación de empleados de forma masiva
- Creación de los siguientes reportes:
    - Reporte de consolidado de planilla Empleados
    - Reporte de consolidado de planilla patronal
    - Reporte de Empleados activos
    - Reporte de consolidado/ Detalle empleado
    - Reporte de deducciones por empresa
    - Reporte de boletas de Pagos
- Tipos de vacaciones (Vacaciones, Permiso con Goce de sueldo, Permiso a cuenta de vacaciones)

8.1.3 Personal involucrado

Nombre	Erving José Salazar Tejeda
Rol	Programador en ASP.Net
Categoría profesional	Ingeniero en Sistemas de la Información
Responsabilidades	Recopilación de Requisitos, Análisis y diseño de Software.
Información de contacto	8751 – 6130 / salazartejeda93@gmail.com
Aprobación	

Nombre	Luis Manuel Oviedo Acosta
Rol	Diseño de Bases de Datos en SQL Server / Consultora de Software
Categoría profesional	Ingeniera en Sistemas de la Información
Responsabilidades	Diseño de la BBDD en SQL Server 2008 R2 y control de calidad de la aplicación
Información de contacto	87619958 / luisoviedoacosta@outlook.es
Aprobación	

Nombre	FRANKLIN ANTONIO ALTAMIRANO ARBIZU
Rol	Miembro del equipo desarrollador
Categoría profesional	Ingeniera en Sistemas de la Información mantenimiento
Responsabilidades	Recopilación de Requisitos, Análisis y diseño de Software.
Información de contacto	89851146 / afranklin057@gmail.com

Nombre	Ing. Juan Carlos Antón Soto
Rol	**Analista de Software / Revisión de cada versión preliminar y definitiva de los módulos del programa**
Categoría profesional	**Master en Análisis y Diseño de Sistemas de Información**
Responsabilidades	**Revisión de los módulos de la aplicación, soporte en todas las fases de la creación del sistema**
Información de contacto	**jcantonsoto@gmail.com**
Aprobación	

8.2 Definiciones, Acrónimos y Abreviaturas

8.2.1 De la empresa

a) **Creación de usuarios:** Proceso por él se crea un usuario con todos sus datos personales, y con su nivel de acceso al sistema.

b) **Publicar Información: Proceso** en el que el usuario ve toda su información personal.

c) **Consultar Información:** Proceso en el que una persona busca información en el SIS-WEB tales como (Boletas de pago, vacaciones, permisos)

d) **Deshabilitar usuario:** Proceso en el que el administrador deshabilita un usuario

e) **Cambio de usuario**: Proceso en el que el administrador o administradores cambian a un usuario su nivel de acceso.

f) **Administrador del sistema:** Persona responsable por la administración del sistema.

g) **Coordinador de Recursos:** Persona responsable de la administración de recursos audiovisuales.

8.2.2 Del Sistema

a) **Administrar:** Acción de agregar, modificar, eliminar y consultar la información de un determinado objeto o persona.

b) **Usuario:** Persona que puede ingresar a la zona del SISTEMA-WEB, a través de un proceso de autenticación utilizando nombre de usuario y contraseña.

c) **Rendimiento:** Referencia al resultado deseado efectivamente obtenido por cada unidad que realiza la actividad, donde el término unidad puede referirse a un individuo, un equipo, un departamento o una sección de una organización.

d) **Permiso: P**arámetro que especifica si su poseedor dispone de acceso a una determinada función del sistema o a una parte de la interfaz de usuario del sistema

e) **Rol:** Es un conjunto de permisos que puede asignarse a un usuario.

f) **Administrador del Sistema:** Persona encargada de ofrecer el soporte técnico y operativo al SISTEMA-WEB.

g) **Pruebas:** Proceso mediante el cual se realizan actividades para verificar la óptima función del sistema.

8.2.3 De tecnología

a) **Dominio:** Nombre base que agrupa a un conjunto de equipos o dispositivos y que permite proporcionar nombres de equipo más fácilmente recordables en lugar de una dirección IP numérica,

b) **Servidor WEB:** Es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.

c) **URL:** Significa Uniform Resource Locator, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

d) **HTTP:** El protocolo de transferencia de hipertexto (HTTP, Hypertext Transfer Protocol) es el protocolo usado en cada transacción de la Web (WWW).

e) **Enlaces:** una referencia en un documento de hipertexto a otro documento o recurso

f) **Sistema de gestión de Base de Datos:** Son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

g) **Base de Datos: Es** un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.

h) **Aplicación WEB: Es** un programa informático diseñado para facilitar al usuario la realización de un determinado tipo de trabajo.

i) **SQL (**Lenguaje de Consulta Estructurado): la cual unifica a un prototipo de lenguaje enlazado con la gestión de bases de datos de condición relacional, que admite la especificación de distintas clases de operaciones entre éstas. Gracias a la utilización del álgebra y de cálculos relacionales.

j) **Dirección IP (Internet Protocol):** Número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo (habitualmente una computadora) dentro de una red que utilice el protocolo IP (Internet Protocol).

k) **Protocolo IP**: El Protocolo de Internet (IP, de sus siglas en inglés Internet Protocol) es un protocolo no orientado a la conexión, usado tanto por el origen como por el destino para la comunicación de estos a través de una red (Internet).

l) **Browser o Motor de Navegación:** Es una aplicación que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web de todo el mundo a través de Internet

m) **DNS:** Es una base de datos distribuida y jerárquica que almacena información asociada de una dirección IP a nombres de dominio en redes como Internet.

n) **ASP.NET (**Hypertext Markup Language**):** Es una tecnología gratuita que permite a los programadores crear páginas web dinámicas, desde website personales hasta aplicaciones web empresariales.

o) **IIS (**servicios de Internet Information**):** son los servicios software que admiten la creación, configuración y administración de sitios web, además de permitir otras funciones de Internet.

p) **Encriptación:** La encriptación es el proceso para volver ilegible información considera importante. Se trata de una medida de seguridad que es usada para almacenar o transferir información delicada que no debería ser accesible a terceros.

q) **Ajax** (Asynchronous JavaScript And XML)**:** es una técnica de desarrollo web para crear aplicaciones interactivas.

r) **RIA (**Rich Internet Applications): es un nuevo tipo de aplicación con más ventajas que las tradicionales aplicaciones Web. Esta surge como una combinación de las ventajas que ofrecen las aplicaciones Web y las aplicaciones tradicionales.

s) **JavaScript:** Es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web.

t) **Internet:** Es un método de interconexión descentralizada de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP

8.3 Referencias

IEEE: Recommended Practice for Software Requirements Specification. ANSI/IEEE std. 830, 1998.

8.4 Descripción general

8.4.1 Perspectiva del producto.

El sistema SIS-WEB será un producto diseñado para trabajar en entornos WEB, lo que permitirá su utilización de forma rápida y eficaz, además trabajará de manera independiente por lo tanto no interactuará con otros sistemas.

8.4.2 Funcionalidad del producto.

El sistema SIS-WEB permitirá realizar las siguientes funciones:

a. **Administración de empleados:** El administrador del sistema podrá gestionar los usuarios (Datos personales, laborales).

b. **Administrar usuarios y sus roles:** El administrador podrá darle acceso al sistema y sus funciones de acuerdo a su perfil de empleado (Gerente, empleado, ejecutivo, etc.).

c. **Cambiar contraseñas y deshabilitar usuarios:** En este caso, solamente el usuario administrador tendrá la capacidad de actualizar y deshabilitar usuarios para todos los usuarios registrados incluyéndose el mismo.

d. **Administración de Roles:** El administrador del sistema gestionará los roles de usuarios del sistema, de igual forma asignará los roles a los usuarios.

e. **Publicación de Información:** Todos los usuarios tendrán acceso a su información Personal y algunas características del sistema de acuerdo a su rol en el SISTEMA-WEB.

g. **Navegación:** Proceso por el cual una persona podrá utilizar el SIS-WEB.

8.4.3 Características de usuario.

Los usuarios de este sistema están familiarizados con el uso de aplicaciones de este tipo ya que simula un formulario normal, que todos conocemos. Es por ello que el sistema ha de ser gráfico. Con una interfaz sencilla, que no exija gran tiempo para su aprendizaje, a la vez que lo suficientemente potente para que el usuario pueda hacer aquello que necesita.

8.5 Perfiles de usuarios

Administrador: es el encargado de instalar o mantener un sistema compuesto por varios componentes informáticos, en óptimas condiciones de funcionamiento para su uso correcto por parte de los usuarios de estos. Para ello debe supervisar el estado del sistema periódicamente, realizar tareas de mantenimiento y de reparación, compra de nuevos elementos, actualizaciones, seguridad, configuración de los equipos y de la red... etc.

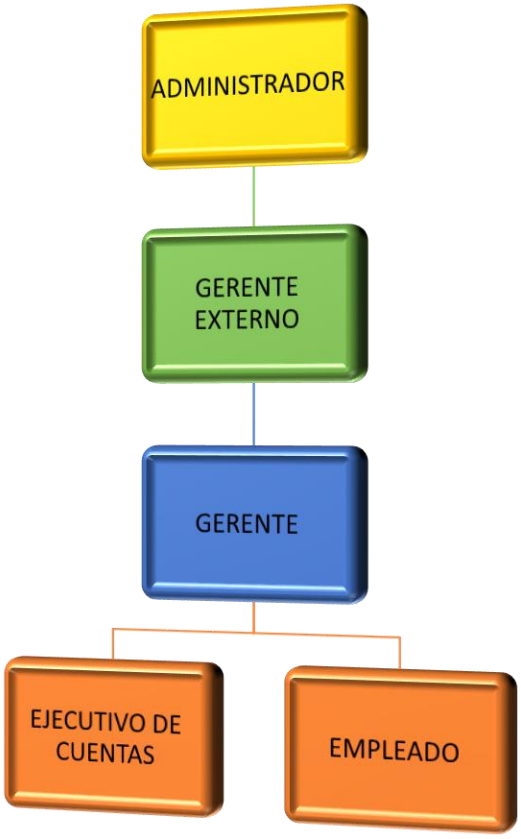
Empleado: Cada empleado es responsable de conocer sus funciones y tareas asignadas, pero también el jefe o responsable debe asegurarse de que el trabajador haya llegado a dominar las habilidades necesarias para realizar sus tareas por completo.

Ejecutivo de Cuentas: es el ejecutivo a cargo de la gestión financiera de la empresa. Es responsable de la planificación, ejecución e información financieras. Generalmente reporta directamente al director gerente de la empresa.

Gerente: es una persona que se encuentra a cargo de la coordinación o dirección de las actividades y funciones de una parte, de un departamento, sector o grupo de trabajo.

Gerente Externo: coordina los recursos internos, representa a la compañía frente a terceros y controla las metas y objetivos.

8.5.1 Jerarquía de usuarios



ADMINISTRA

GERENTE

EJECUTIVO DE CUENTAS

EMPLEADO

Tipo de usuario	Administrador
Formación	Programador
Habilidades	Control total de manejar el Sistema web
Actividades	Ingresara usuarios; actualizara el sistema

Tipo de usuario	Usuarios
Formación	Bachiller
Habilidades	Manejo básico en programas de informática
Activities	Completar el formulario para programar su actividad a realizar.

Tipo de usuario	Usuarios Administradores
Formación	Licenciatura o Programador
Habilidades	Control total o parcial del Sistema de acuerdo a sus roles asignados
Actividades	Encargado de estar en contacto con los usuarios

8.6 Restricciones.

El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma o del lenguaje de programación. El sistema se diseñará según un modelo cliente servidor.

8.6.1 Políticas reguladoras

La aplicación se desarrollará mediante software de licencia por lo tanto se deberá pagar por el uso de: servidor WEB (IIS), Sistema de Gestión de base de datos (SQL Server) y el lenguaje de programación (ASP.NET), por lo tanto, la utilización de estos programas se hará mediante las políticas establecidas por este tipo de licenciamiento.

8.6.2 Limitaciones de hardware

Para esta aplicación será necesario un computador servidor en el cual se instalará el servidor WEB IIS, SQL Server y la aplicación SIS-WEB.

8.6.3 Interfaces con otras aplicaciones

Debido a que el sistema no interactúa con otros sistemas y es autónomo no se desarrollaran interfaces con otras aplicaciones. Las conexiones necesarias para la utilización del servidor web, SQL y un DNS, se hará por medio de la configuración de estos programas.

8.6.4 Funciones de control

El sistema debe controlar los permisos que tiene cada usuario para su accesibilidad de una manera correcta, de tal forma que pueda acceder la información que le corresponde de acuerdo a su rol. Debe tener controles adecuados para la validación de datos, de igual manera la programación de las actividades específicas para cada usuario.

8.7 Requisitos específicos

8.7.1 Interfaces Externas.

El sistema SISTEMA-WEB tendrá interconexión con otros sistemas de información, por lo tanto, es necesario la utilización de una interfaz.

La relación con un servidor WEB, DNS y Gestor de Base de Datos se hará a través de los archivos de configuración de éstos.

8.7.2 Requerimientos Funcionales

Identificación del requerimiento:	RF01
Nombre del Requerimiento:	Autentificación de Usuario.
Características:	Los usuarios deberán identificarse para acceder a cualquier parte del sistema.
Descripción del requerimiento:	El sistema podrá ser consultado por cualquier usuario dependiendo del módulo en el cual se encuentre y su nivel de accesibilidad.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF02
Nombre del Requerimiento:	Creación de nuevas empresas dentro del sistema.
Características:	Las creaciones de nuevas empresas estarán a cargo por el usuario administrador.
Descripción del requerimiento:	El sistema permitirá al Administrador registrar nuevas empresas dentro del sistema y este les asignara el nivel de acceso a los usuarios correspondientes.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF03
Nombre del Requerimiento:	Creación de los departamentos de área
Características:	El administrador creará los departamentos de áreas por empresa.
Descripción del requerimiento:	El administrador creará los departamentos de áreas de trabajos para organizar a los empleados de acorde a sus labores y se le asignaran a la empresa correspondiente.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF04
Nombre del Requerimiento:	Creación de puestos por departamento.
Características:	El administrador creara el puesto correspondiente a cada departamento.
Descripción del requerimiento:	El administrador crear el puesto y se lo asignará a un determinado departamento de área.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF5
Nombre del Requerimiento:	Creaciones de grupos de empresas
Características:	El administrador crea en nombre del grupo de la empresa.
Descripción del requerimiento:	El administrador crea el nombre del grupo de la empresa y después se asignan las empresas correspondientes a ese grupo.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF6
Nombre del Requerimiento:	Administración de Empleados
Características:	Creación de registro de cada empleado.
Descripción del requerimiento:	El sistemas permitirá al administrador registrar los dato personales de cada empleado, también podrá actualizar la información de cada una de ellos.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF7
Nombre del Requerimiento:	Administración de Usuario
Características:	El sistema permitirá al administrador ingresar los datos del usuario.
Descripción del requerimiento:	El sistema permitirá al administrador ingresar y manipular los datos del usuario y también podrá definir si perfil de usuario el cual tendrá los privilegios de acceso al sistema. Dentro de los perfiles de usuario están: Administrador, Ejecutivo de cuenta, Gerente y empleados.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF8
Nombre del Requerimiento:	Administrar planillas
Características:	El Ejecutivo de cuenta creara y administrara las planillas de pago que se procesan en la empresa.
Descripción del requerimiento:	El sistema permitirá al Ejecutivo crear el nombre de la planilla, el formato y luego procederá importar los datos desde un Excel para poder guardarla en la base de datos.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF9
Nombre del Requerimiento:	Importación de los empleados de forma masiva
Características:	El administrador podrá importar los datos de los empleados desde un Excel para guardarlos en la base de datos.
Descripción del requerimiento:	El sistema permitirá al administrador importar empleados de forma masivas desde un Excel con formato determinado conforme a la estructura de la base de datos
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF10
Nombre del Requerimiento:	Gestionar Reportes
Características:	El sistema permitirá generar reportes.
Descripción del requerimiento:	Permite al administrador imprimir todos los reportes de las planillas de lo empleado y también cada usuario puede ver su propio reporte de su comprobante de pago. Los reportes que se pueden mostrar son de planillas de pagos, boletas de pagos, detalles de los empleados, empleados activos e inactivos y consolidados de planillas de empleados y patronal de deducciones.
Prioridad del requerimiento:	Alta

Identificación del requerimiento:	RF11
Nombre del Requerimiento:	Solicitud de permisos
Características:	El sistema permitirá solicitar permisos a los empleados
Descripción del requerimiento:	El sistema permite a los empleados solicitar permiso de vacaciones, abra permisos con goce de sueldos y permisos sin goce de sueldo.
Prioridad del requerimiento: Alta	

8.7.3 Requerimientos no Funcionales.

Identificación del requerimiento:	RNF01
Nombre del Requerimiento:	Interfaz del sistema.
Características:	El sistema presentara una interfaz de usuario sencilla para que sea de fácil manejo a los usuarios del sistema.
Descripción del requerimiento:	El sistema debe tener una interfaz de uso intuitiva y sencilla.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF02
Nombre del Requerimiento:	Ayuda en el uso del sistema.
Características:	La interfaz del usuario deberá de presentar un sistema de ayuda para que los mismos usuarios del sistema se les faciliten el trabajo en cuanto al manejo del sistema.
Descripción del requerimiento:	La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas).
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF03
Nombre del Requerimiento:	Mantenimiento.
Características:	El sistema deberá de tener un manual de instalación y manual de usuario para facilitar los mantenimientos que serán realizados por el administrador.
Descripción del requerimiento:	El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF04
Nombre del Requerimiento:	Diseño de la interfaz a la característica de la web.
Características:	El sistema deberá de tener una interfaz de usuario, teniendo en cuenta las características de la web de la institución.
Descripción del requerimiento:	La interfaz de usuario debe ajustarse a las características de la web de la institución, dentro de la cual estará incorporado el sistema de gestión de procesos y el inventario.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF05
Nombre del Requerimiento:	Desempeño
Características:	El sistema garantizara a los usuarios un desempeño en cuanto a los datos almacenado en el sistema ofreciéndole una confiabilidad a esta misma.
Descripción del requerimiento:	Garantizar el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF06
Nombre del Requerimiento:	Nivel de Usuario
Características:	Garantizara al usuario el acceso de información de acuerdo al nivel que posee.
Descripción del requerimiento:	Facilidades y controles para permitir el acceso a la información al personal autorizado a través de Internet, con la intención de consultar y subir información pertinente para cada una de ellas.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF07
Nombre del Requerimiento:	Confiabilidad continúa del sistema.
Características:	El sistema tendrá que estar en funcionamiento las 24 horas los 7 días de la semana. Ya que es una página web diseñada para la carga de datos y comunicación entre usuarios.
Descripción del requerimiento:	La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF08
Nombre del Requerimiento:	Seguridad en información
Características:	El sistema garantizara a los usuarios una seguridad en cuanto a la información que se procede en el sistema.
Descripción del requerimiento:	Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
Prioridad del requerimiento: Alta	

8.8 Requisitos comunes de las interfaces

8.8.1 Interfaces de usuario

La interfaz con el usuario consistirá en un conjunto de ventanas con botones, listas y campos de textos. Ésta deberá ser construida específicamente para el sistema propuesto y, será visualizada desde un navegador de internet.

8.8.2 Interfaces de hardware

Será necesario disponer de equipos de cómputos en perfecto estado con las siguientes características:

- Adaptadores de red.
- Procesador de 1.10GHz o superior.
- Memoria RAM mínima de 2BG.
- Mouse.
- Teclado.

8.8.3 Interfaces de software

- Sistema Operativo: Windows 7 o superior.
- Explorador: Mozilla o Chrome y otros.

8.8.4 Interfaces de comunicación

Los servidores, clientes y aplicaciones se comunicarán entre sí, mediante protocolos estándares en internet, siempre que sea posible. Por ejemplo, para transferir archivos o documentos deberán utilizarse protocolos existentes (FTP u otros convenientes).

8.9 Requisitos funcionales

8.9.1 Requisito funcional 1

8.9.1.1 Autentificación de Usuario

- **Introducción:** Los usuarios registrados podrán ingresar al sistema para hacer uso de sus funciones.
- **Entradas:** Los usuarios deberán proporcionar al sistema la siguiente información: Nombre y Contraseña de usuario.
- **Procesos:** Se validarán los datos introducidos y si son correctos se procederá a ingresar al sistema
- **Salidas:** Si la información ingresada es válida se habilitarán los controles correspondientes al nivel del usuario, de lo contrario se mostrará un mensaje indicando que los datos no son válidos.

8.9.2 Requisito funcional 2

8.9.2.1 Creación de nuevas Empresas

- **Introducción:** Este proceso deberá realizar la captura por terminal de todos los datos pertenecientes a la empresa.
- **Entradas:** Por pantalla
    - **Código empresa:** Dato obligatorio que identificara a la empresa
    - **Nombre:** Dato obligatoria que identifica a la empresa como marca comercial.
    - **RUC:** Dato obligatorio almacena el número correspondiente a la empresa
    - **Teléfono:** Dato obligatorio almacena el número de teléfono.
    - **Dirección:** Dato obligatorio que almacena la dirección de la empresa.
    - **País:** Dato obligatorio que identifica el país de origen de la empresa.
    - **Ciudad**: Dato obligatorio que identifica en la ciudad.
    - **Grupo de empresa:** Dato obligatorio almacena el grupo de empresa a que pertenece
        - **Correo electrónico:** Dato obligatorio que almacena el correo de la empresa
- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:
    - **Código empresa**
    - **Nombre**
    - **RUC**
    - **Teléfono:**
    - **Dirección**
    - **País**
    - **Ciudad**:
    - **Grupo de empresa**
    - **Correo electrónico**

Para actualizar los datos introducidos seleccionamos la empresa y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a guardar el registro con la nueva empresa.

8.9.3 Requisito funcional 3

8.9.3.1 Creación de departamento de áreas

- **Introducción:** Este proceso deberá realizar la captura de departamentos de área por empresa.
- **Entradas:** Por pantalla
  - **Nombre del departamento:** Dato obligatorio que identificara el departamento.
  - **Nombre de empresa:** Dato obligatoria que identifica a la empresa que pertenecerá el departamento.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:
  - **Nombre de departamento**
  - **Nombre de empresa**

Para actualizar los datos introducidos seleccionamos el nombre de la empresa y se nos desplegara todos los departamentos asignada a esa empresa y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a guardar el registro con la nueva empresa.

8.9.4 Requisito funcional 4

8.9.4.1 Creación de puesto por departamento

- **Introducción:** Este proceso deberá realizar la captura de puestos de trabajo por departamento de área.
- **Entradas:** Por pantalla
    - **Nombre del puesto de trabajo:** Dato obligatorio que identificara el nombre del puesto de trabajo.
    - **Nombre del departamento:** Dato obligatorio que identificara el departamento.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:
    - **Nombre del puesto de trabajo**
    - **Nombre de los departamentos**

Para actualizar los datos introducidos seleccionamos el nombre del departamento y se nos desplegara todos los puestos de trabajo asignado a ese departamento y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a guardar el registro con el nuevo puesto de trabajo.

8.9.5 Requisito funcional 5

8.9.5.1 Creación de grupos de empresas

- **Introducción:** Este proceso deberá realizar la captura de un grupo empresarial.
- **Entradas:** Por pantalla
    - **Nombre del grupo empresarial:** Dato obligatorio que identificara el nombre del grupo empresarial.
- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:

    - **Nombre del grupo empresarial**

Para actualizar los datos introducidos seleccionamos el grupo empresarial y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a guardar el registro como un nuevo grupo empresarial.

8.9.6 Requisito funcional 6

8.9.6.1 Administración de empleados

- **Introducción:** Este proceso deberá realizar la captura de un grupo empresarial.
- **Entradas:** Por pantalla

- **Código de empleado:** Dato opcional Código de identificación dentro de la empresa
- **Primer Nombre:** Dato obligatorio que identificara el primer nombre del empleado.
- **Segundo Nombre:** Dato no obligatorio que identificara el segundo nombre del empleado.
- **Primer Apellido:** Dato obligatorio que identificara el primer apellido del empleado.
- **Segundo Apellido:** Dato no obligatorio que identificara el segundo apellido del empleado.
- **Número de identificación:** Dato obligatorio que identificara al empleado con su número de Cedula/Pasaporte (Extranjero).
- **Nombre del conyugue:** Dato no obligatorio que identificara el nombre del grupo empresarial.
- **Celular:** Dato no obligatorio que identificara número de celular del empleado.
- **País:** Dato obligatorio que identificara el país de origen/nacimiento.
- **Nacionalidad:** Dato obligatorio que identificara la nacionalidad.
- **Estado civil:** Dato obligatorio que identificara el estado civil del empleado
- **Correo:** Dato obligatorio que de su correo electrónico.
- **Genero** Dato obligatorio que identificara el sexo de la persona
- **Dirección:** Dato obligatorio que identificara la dirección en que reside el empleado
- **Fecha de nacimiento:** Dato obligatorio que identificara su fecha de nacimiento.
- **Empresa:** Dato obligatorio que identificara la empresa donde laborara.
- **Grupo Empresa:** Dato obligatorio que identificara el grupo empresarial.

- **Departamento:** Dato obligatorio que identifica el departamento en laborara.
- **Puesto:** Dato obligatorio el puesto que desempeñara dentro de la empresa.
- **Salario:** Dato obligatorio que identificara el salario que devengara por su trabajo.
- **Jefe Inmediato:** Dato obligatorio que identificara al superior que se reportara.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:

  - **Código de empleado**
  - **Primer Nombre**
  - **Segundo Nombre**
  - **Primer Apellido**
  - **Segundo Apellido**
  - **Número de identificación**
  - **Nombre del conyugue**
  - **Celular**
  - **País**
  - **Nacionalidad**
  - **Estado civil**
  - **Correo**
  - **Genero**
  - **Dirección**
  - **Fecha de nacimiento**
  - **Empresa:**
  - **Grupo Empresa:**
  - **Departamento:**
  - **Puesto:**

- **Salario:**
- **Jefe Inmediato:**

Los campos Empresa, Grupo de empresa, Departamento, Puesto, Jefe inmediato podrán ser seleccionados de forma selectiva de una lista desplegable.

Para actualizar los datos introducidos seleccionamos el empleado y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a guardar el registro como un nuevo empleado.

8.9.7 Requisito funcional 7

8.9.7.1 Administración de usuarios

- **Introducción:** Este proceso deberá realizar la creación de usuario para el acceso al sistema.
- **Entradas:** Por pantalla
  - **Nombre de usuario:** Dato obligatorio que identificara el nombre de usuario para el acceso al sistema.
  - **Contraseña:** Dato obligatorio que identificara la contraseña del usuario.
  - **Nombre:** Dato obligatorio que identificara el nombre del usuario
  - **Apellido:** Dato obligatorio que identificara el apellido del usuario
  - **Empresa:** Dato obligatorio que identificara la empresa en que labora
  - **Puesto:** Dato obligatorio que identificara el puesto que desempeña
  - **Correo:** Dato obligatorio que identificara el correo del usuario
  - **Teléfono:** Dato obligatorio que identificara el teléfono del usuario

- **Perfil:** Dato obligatorio que identificara el tipo de perfil de usuario, el cual ya tiene determinados permisos de acceso al sistema.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:
  - **Nombre de usuario**
  - **Contraseña**
  - **Nombre**
  - **Apellido**
  - **Empresa**
  - **Puesto**
  - **Correo**
  - **Teléfono**
  - **Perfil**

Los campos Nombre, Apellido, Empresa, Puesto, Correo, Teléfono estarán ya precargados al momento de crear el empleado. El perfil de usuario es una lista desplegable en donde cada uno tiene asignado sus permisos al sistema.

Para actualizar los datos introducidos seleccionamos el nombre del usuario y le damos al botón editar y se habilitan todos los campos anteriormente pedidos y se realiza las modificaciones y se pulsa el botón guardar.

Salidas: Con los datos antes mencionados y ya llenados se procede a guardar el registro con el nuevo usuario del sistema.

8.9.8 Requisito funcional 8

8.9.8.1 Administraciones planillas

- **Introducción:** Este proceso deberá realizar la creación las planillas de empleados y patronales.
- **Entradas:** Por pantalla
    - **Nombre de la planilla:** Dato obligatorio que identificara la planilla.
    - **Nombre de la empresa:** Dato obligatorio que identificara a la empresa que pertenece la planilla.
    - **Fecha de creación:** Dato obligatorio que identifica la fecha de creación de la planilla con su formato.
    - **Numero de Empleado:** Dato obligatorio que identifica el numero empleado en dicha planilla
    - **Factor de cambio:** Dato obligatorio que identifica el valor en dólares.
    - **Moneda:** Dato obligatorio que identifica la moneda de cambio.
    - **Fecha de pago:** Dato obligatorio que identifica la fecha que se pagó la planilla
    - **Fecha de inicio:** Dato obligatorio que identifica la fecha de inicio del pago de la planilla
    - **Fecha final:** Dato obligatorio que identifica la fecha final del pago de la planilla

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:
    - **Nombre de la planilla**
    - **Nombre de la empresa**
    - **Fecha de creación**
    - **Numero de Empleado**
    - **Factor de cambio**
    - **Moneda**

- **Fecha de pago**
- **Fecha de inicio**
- **Fecha final**

Para actualizar los datos introducidos seleccionamos el nombre de la planilla, le damos al botón editar y se habilitan todos los campos anteriormente pedidos, se realiza las modificaciones y se pulsa el botón guardar.

Salidas: Con los datos antes mencionados y ya llenados se procede a guardar la planilla ya con sus datos.

8.9.9 Requisito funcional 9

8.9.9.1 Importación de empleados de forma masiva

- **Introducción:** Este proceso deberá realizar la importación de datos de empleado, así como sus usuarios.
- **Entradas:** Por pantalla
  - **Solicitud de busque da Excel:** Dato obligatorio que identifica archivo con los datos
- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberá buscar el archivo especificado.

Salidas: Con el archivo anteriormente requerido, se da clic en botón importar datos.

8.9.10 Requisito funcional 10

8.9.10.1 Gestionar Reportes Varios

- **Introducción:** Este proceso se crean diversos tipos de reporte dependiendo de la parte interesada.
- **Entradas:** Por pantalla
    - **Fecha de inicio:** Dato obligatorio que identifica la fecha de inicio del reporte.
    - **Fecha de final:** Dato obligatorio que identifica la fecha de final del reporte.
    - **Empresa o grupo empresarial:** Dato obligatorio que identifica la empresa o grupo empresarial del reporte.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberá seleccionar el rango de fechas y la empresa de la que se requiera el reporte.
    - **Fecha de inicio**
    - **Fecha de final**
    - **Empresa o grupo empresarial**

  Salidas: con los datos anteriormente pedidos se pueden obtener diversos tipos de reporte y a su vez ser filtrados dependiendo de la parte interesada. Como son: Planillas de pago, boletas de pago, detalles de los empleados, empleados activos e inactivos, consolidado de planillas de empleado y patronal de deducciones

8.9.11 Requisito funcional 11

8.9.11.1 Solicitud de permiso

- **Introducción:** Este proceso deberá realizar la solicitud de un permiso.
- **Entradas:** Por pantalla

- **Días disponibles:** Dato no obligatorio requerido por el sistema que muestras los días de vacaciones obtenidos.
- **Descripción:** Dato obligatorio que describe el motivo de la solicitud.
- **Fecha de inicio:** Dato obligatorio que identificara la fecha de inicio.
- **Fecha de retorno:** Dato obligatorio que identifica la fecha de regreso o retorno al trabajo.
- **Tipo de solicitud:** Dato obligatorio que identificara el tipo de solicitud.
- **Duración:** Dato no obligatorio requerido por el sistema que muestra la duración del permiso.

- **Procesos:** se mostrará al usuario una interfaz amigable en el cual se deberán ingresar los siguientes campos:

  - **Días disponibles**
  - **Descripción**
  - **Fecha de inicio**
  - **Fecha de retorno**
  - **Tipo de solicitud**
  - **Duración**

Los campos días disponibles se pre-cargara al solicitar el permiso y duración se calcularán automáticamente dentro del sistema.

- **Salidas:** Con los datos antes mencionados y ya llenados se procede a enviar la solicitud a su jefe inmediato y este esperara por la aprobación.

- **Consultar Información:** Muestra información general sobre los privilegios asignados según su tipo de autentificación. El sistema ofrecerá al usuario información general acerca de la Instrucción de la empresa ACAPROF.

- ✓ **Consultar Instrucción Acaprof:** Muestra información general sobre las funcionalidades que ofrece el sistema web a la empresa ACAPROF como: Registrar empleados, agregar departamentos en la empresa, ver planillas de los empleados, esto dependerá de los privilegios asignados a usuario.

8.10 Requisitos no funcionales

8.10.1 Requisitos de rendimiento

- ❖ Garantizar que el diseño de las consultas u otro proceso no afecten el desempeño de la base de datos, ni considerablemente el tráfico de la red.

8.10.1.1 Seguridad

Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados sin que afecte el tiempo de respuesta.

Garantizar la seguridad del sistema con respecto a la información y datos que se manejan como registros, archivos, contraseñas.

8.10.1.2 Fiabilidad

El sistema debe de tener una interfaz de uso intuitiva y sencilla. La interfaz de usuario debe ajustarse a las características de la web de la institución, dentro de la cual estará incorporado el sistema.

8.10.1.3 Disponibilidad

La disponibilidad del sistema debe de ser continua, con un nivel de servicio de 7 días las 24 horas.

8.10.1.4 Mantenibilidad

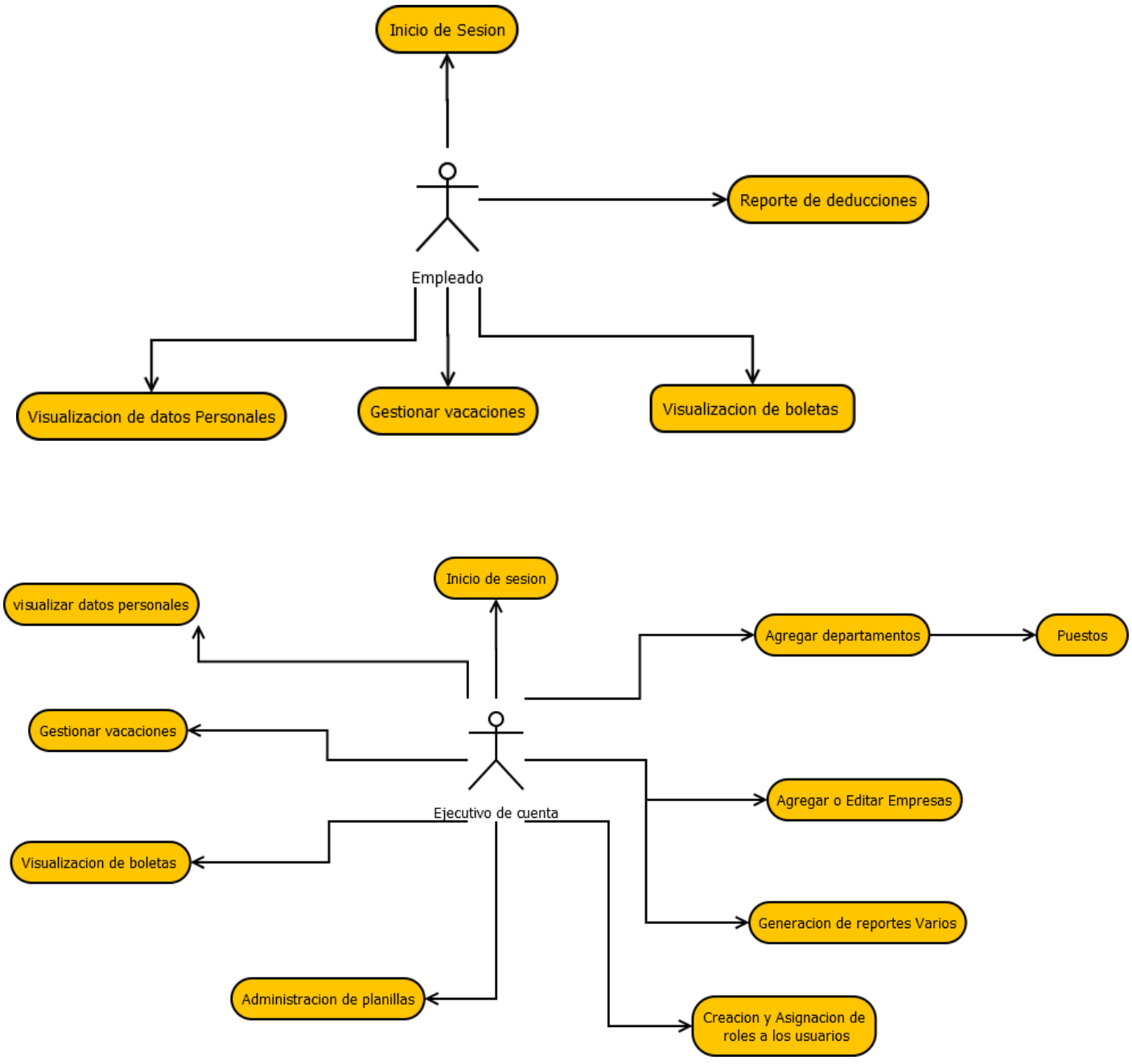
El sistema debe de disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.

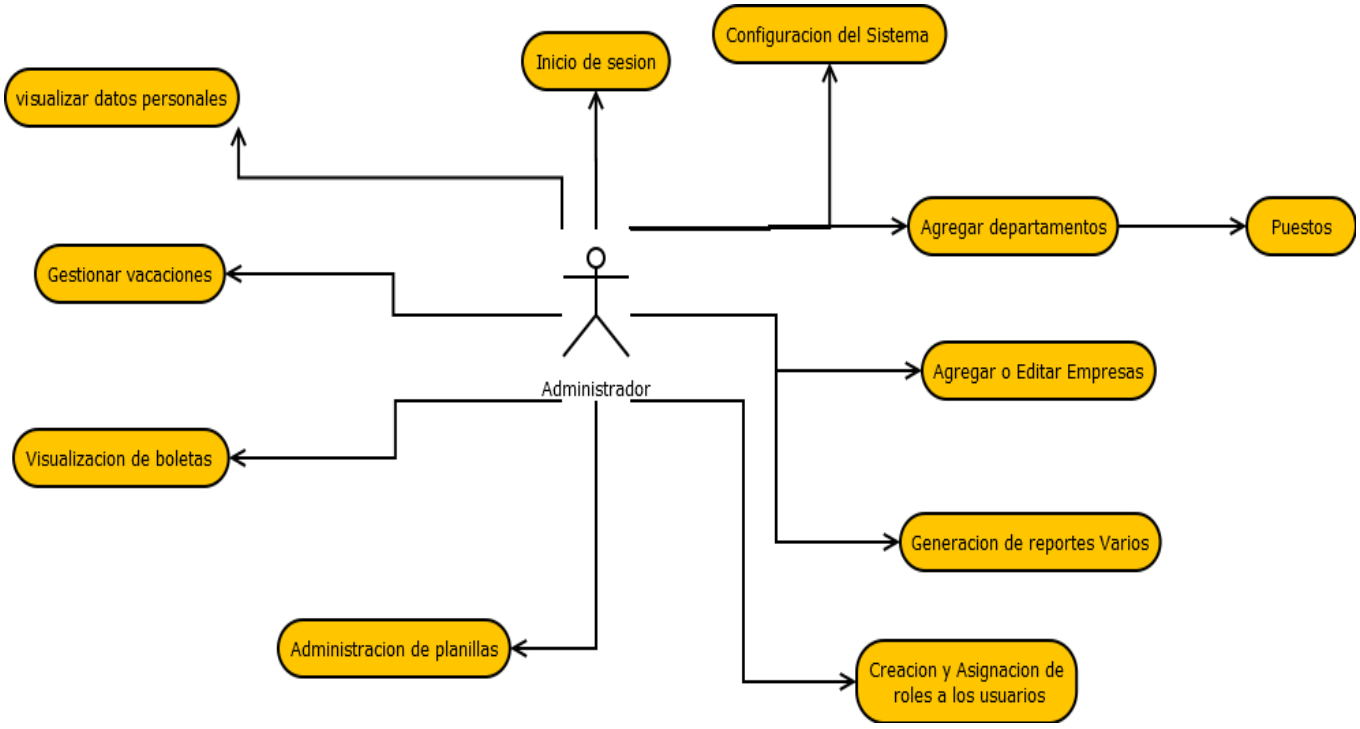
La interfaz puede ser complementada con un buen sistema de ayuda (la administración de este puede recaer en personal de poca experiencia en el uso de aplicaciones informáticas).

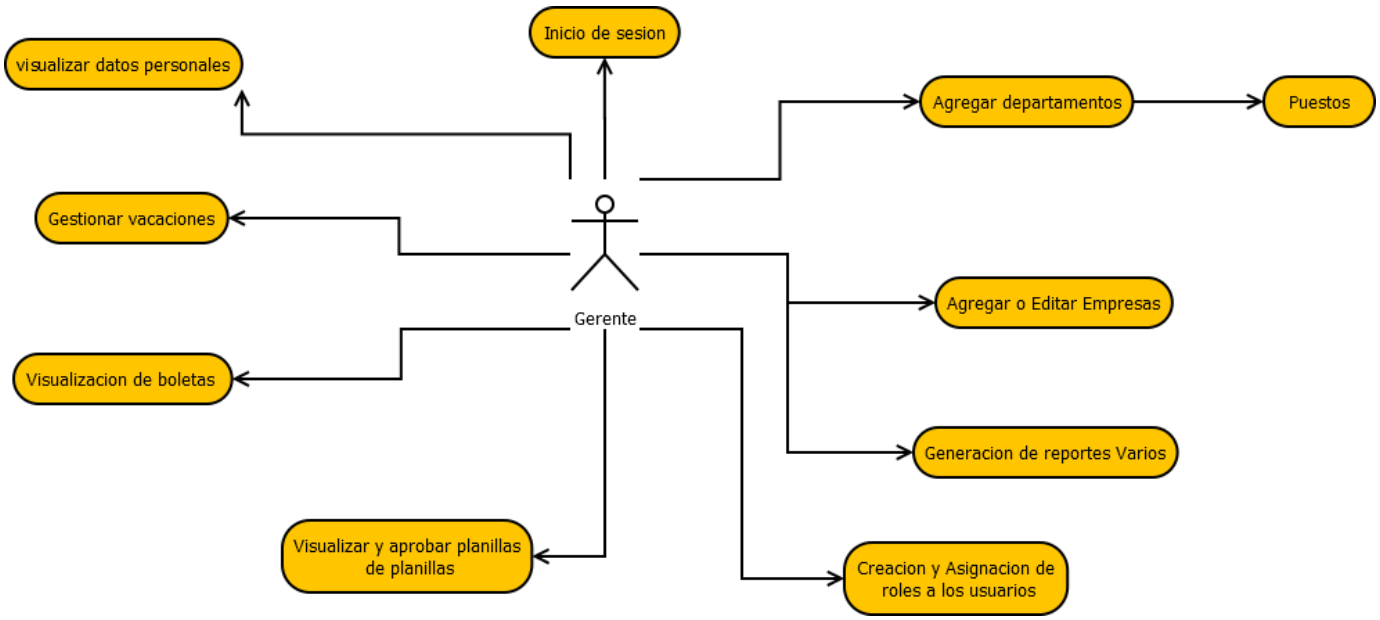
El tiempo de respuesta a consultas, actualizaciones, altas, modificaciones y bajas ha de ser inferior a 10 segundos. Estructura de datos para el almacenamiento de los datos sencilla. Proporcionar la mayor cantidad de información al usuario.

9. CASOS DE USOS

9.1 Diagrama de Casos de Uso




Inicio de sesion
Configuracion del Sistema
visualizar datos personales
Agregar departamentos
Puestos
Gestionar vacaciones
Administrador
Agregar o Editar Empresas
Visualizacion de boletas
Generacion de reportes Varios
Administracion de planillas
Creacion y Asignacion de roles a los usuarios


Inicio de sesion
visualizar datos personales
Agregar departamentos
Puestos
Gestionar vacaciones
Gerente
Agregar o Editar Empresas
Visualizacion de boletas
Generacion de reportes Varios
Visualizar y aprobar planillas de planillas
Creacion y Asignacion de roles a los usuarios

9.2 Diagrama de Secuencia

En esta etapa se abordará la arquitectura del sistema. A continuación, se muestran los diagramas de secuencia.

Diagrama de secuencia 1: Inicio de sesión.

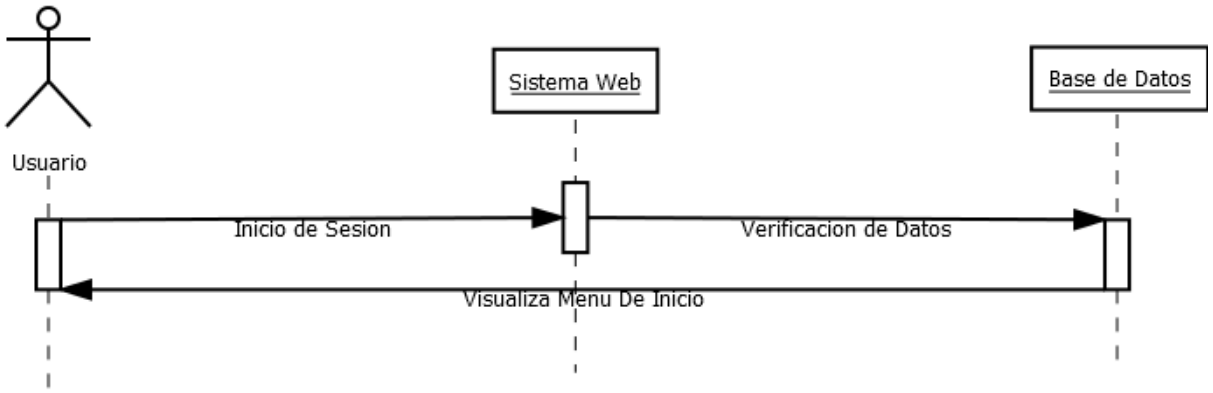


Diagrama de secuencia 2: Usuario administrador creación de Empresa.

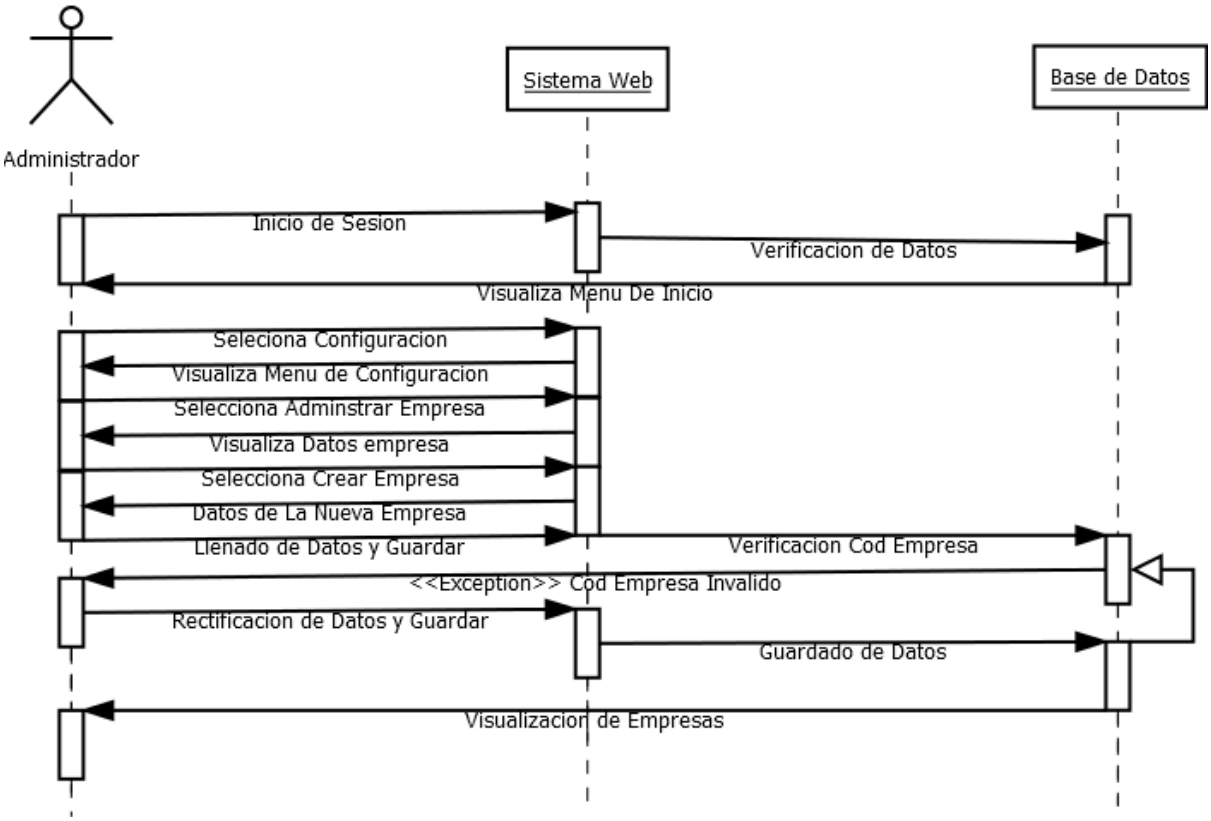


Diagrama de secuencia 3: Creación de Grupos de Empresas.

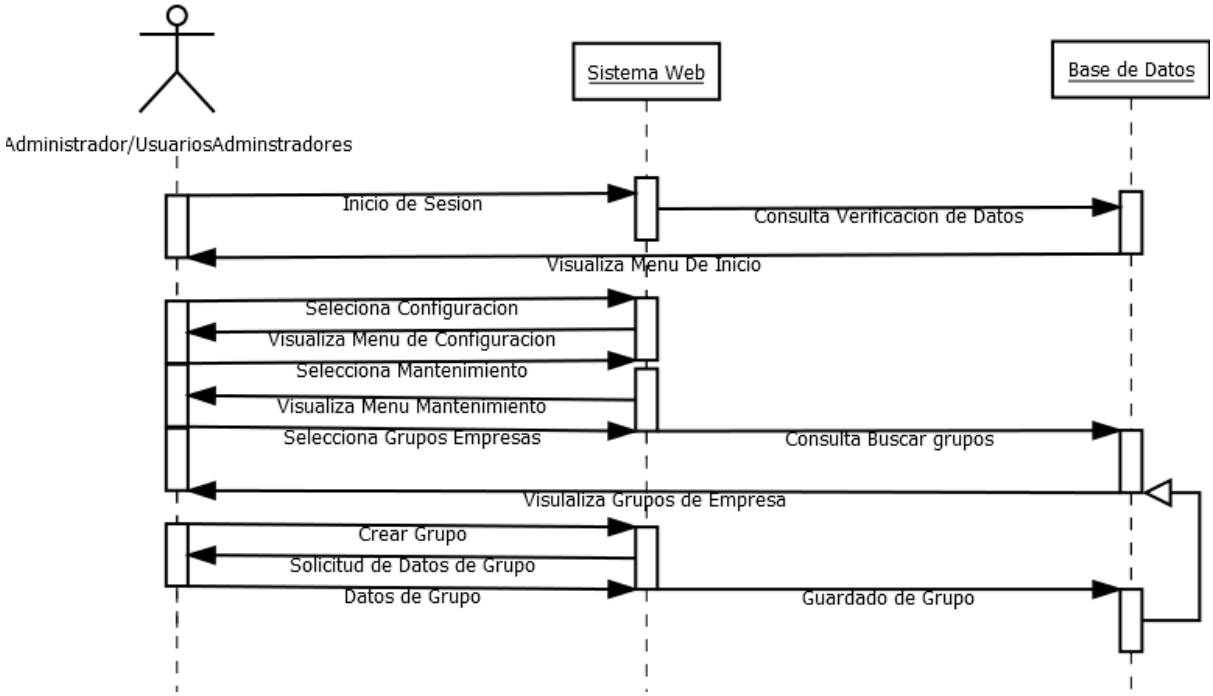


Diagrama de secuencia 4: Creación de Departamentos.

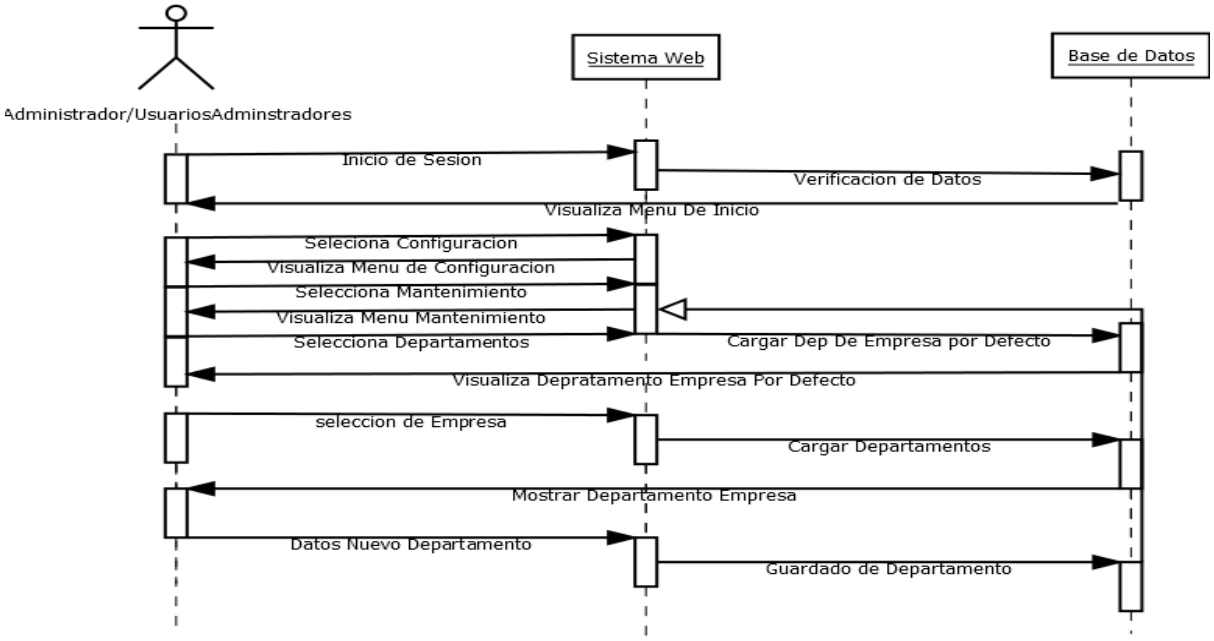


Diagrama de secuencia 5: Creación de Puestos.

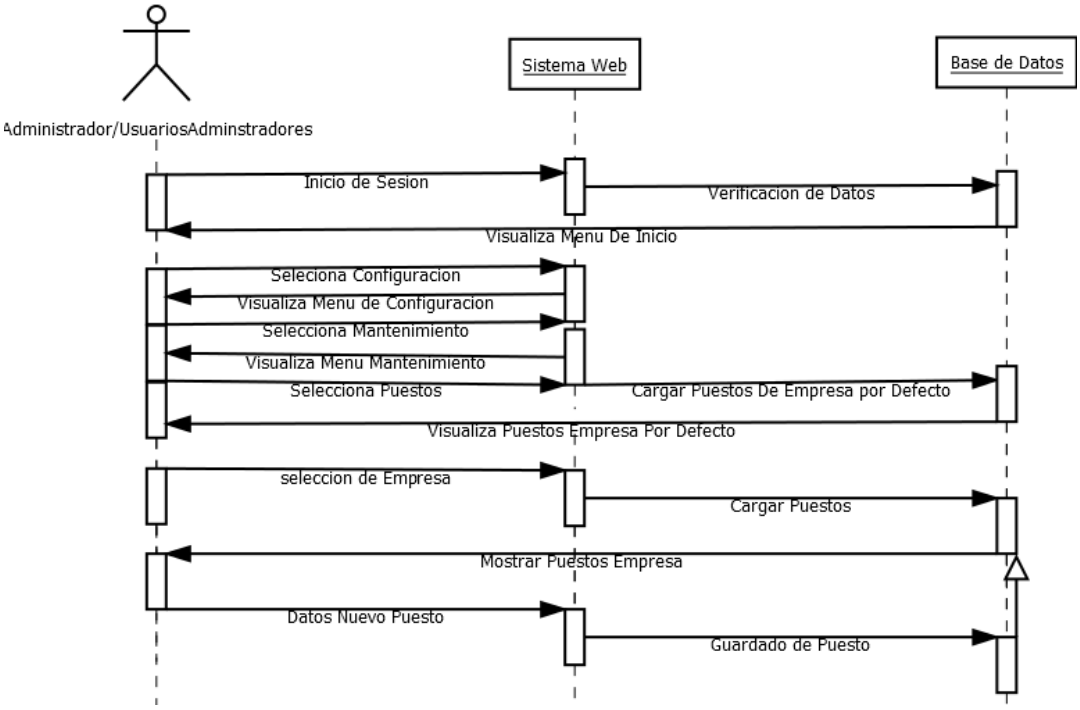


Diagrama de secuencia 6: Creación de Empleados.

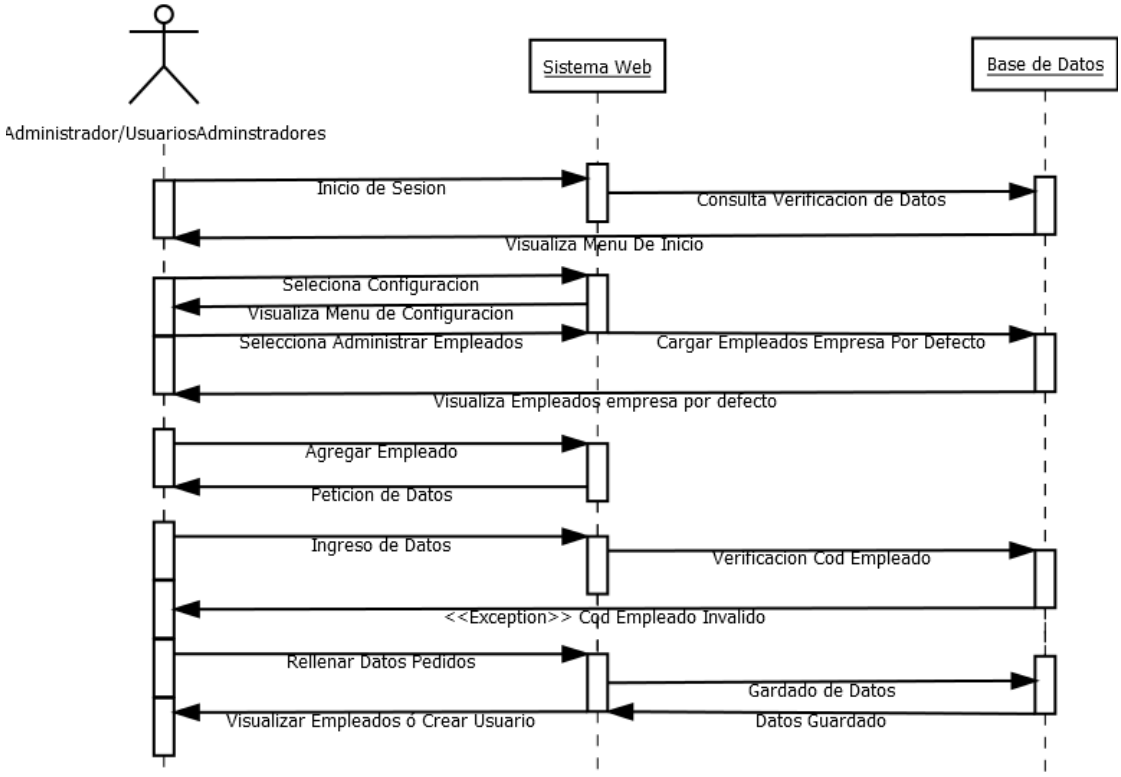


Diagrama de secuencia 7: Creación de Usuarios.

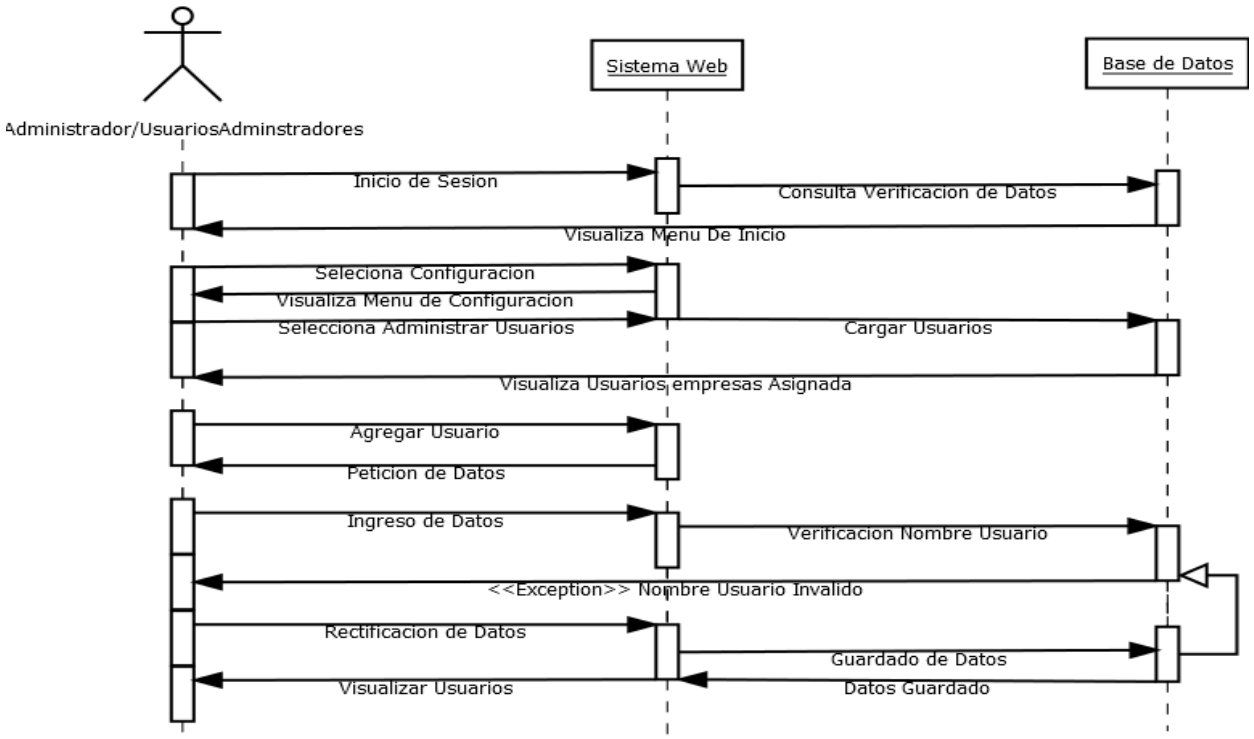


Diagrama de secuencia 8: Administrar Planillas.

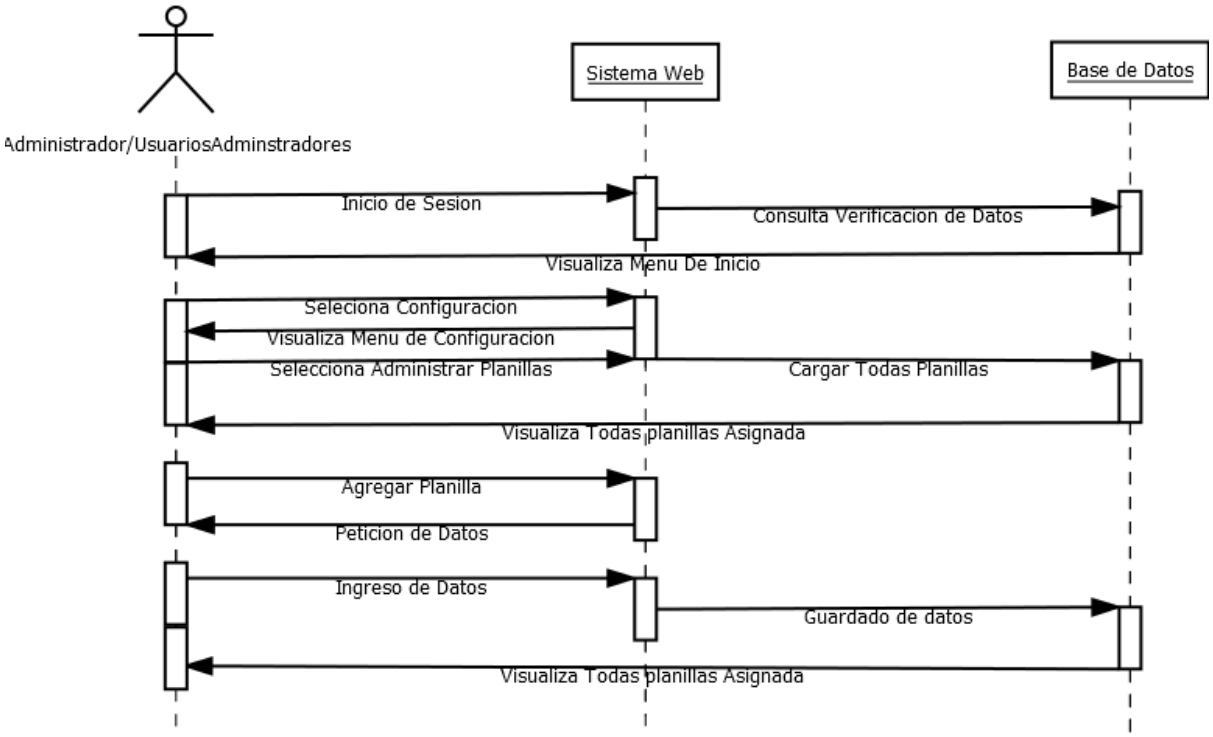


Diagrama de secuencia 9: Importación de Empleados.

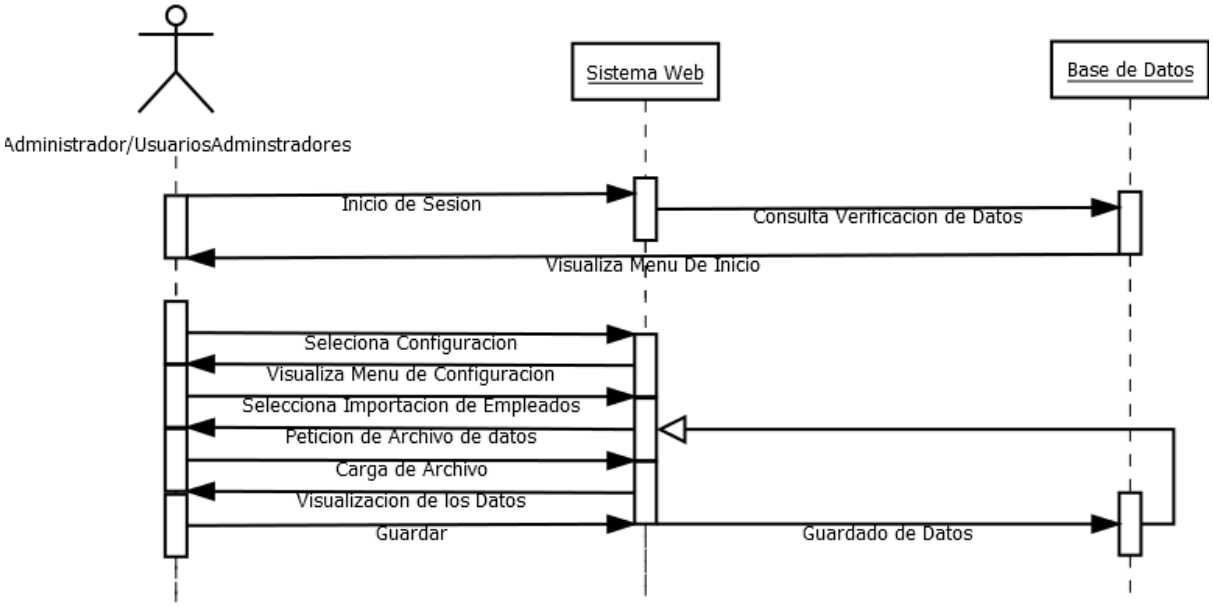


Diagrama de secuencia 10: Creación de Reportes.

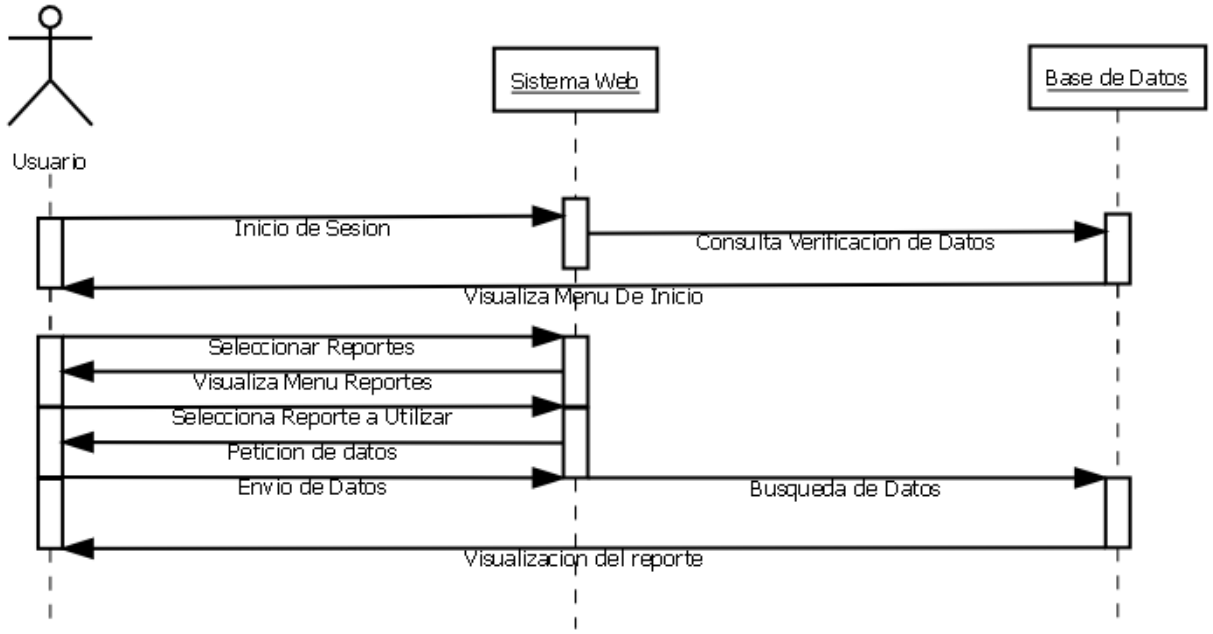
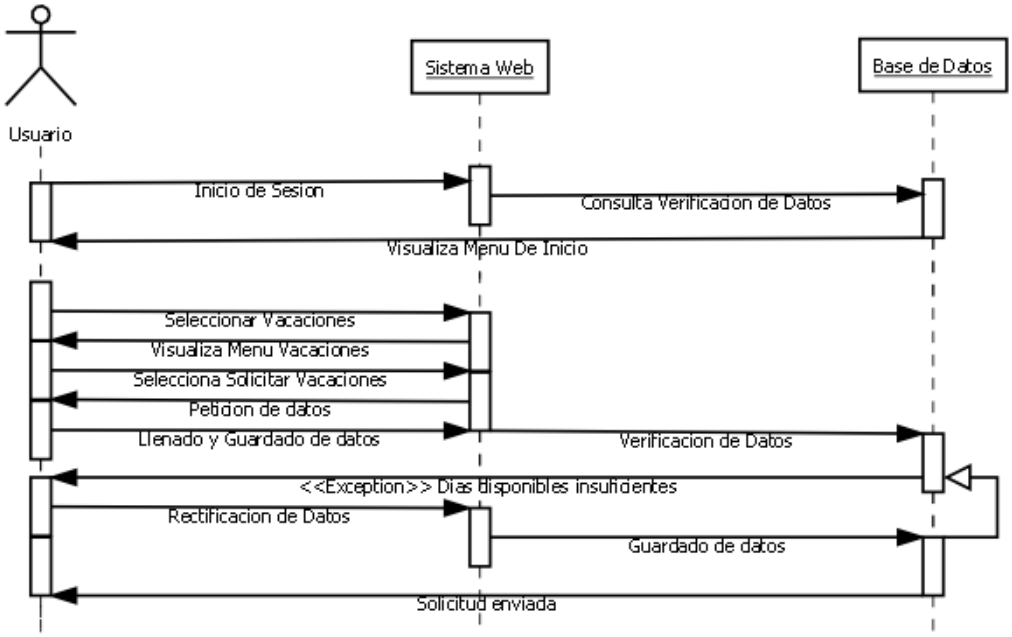
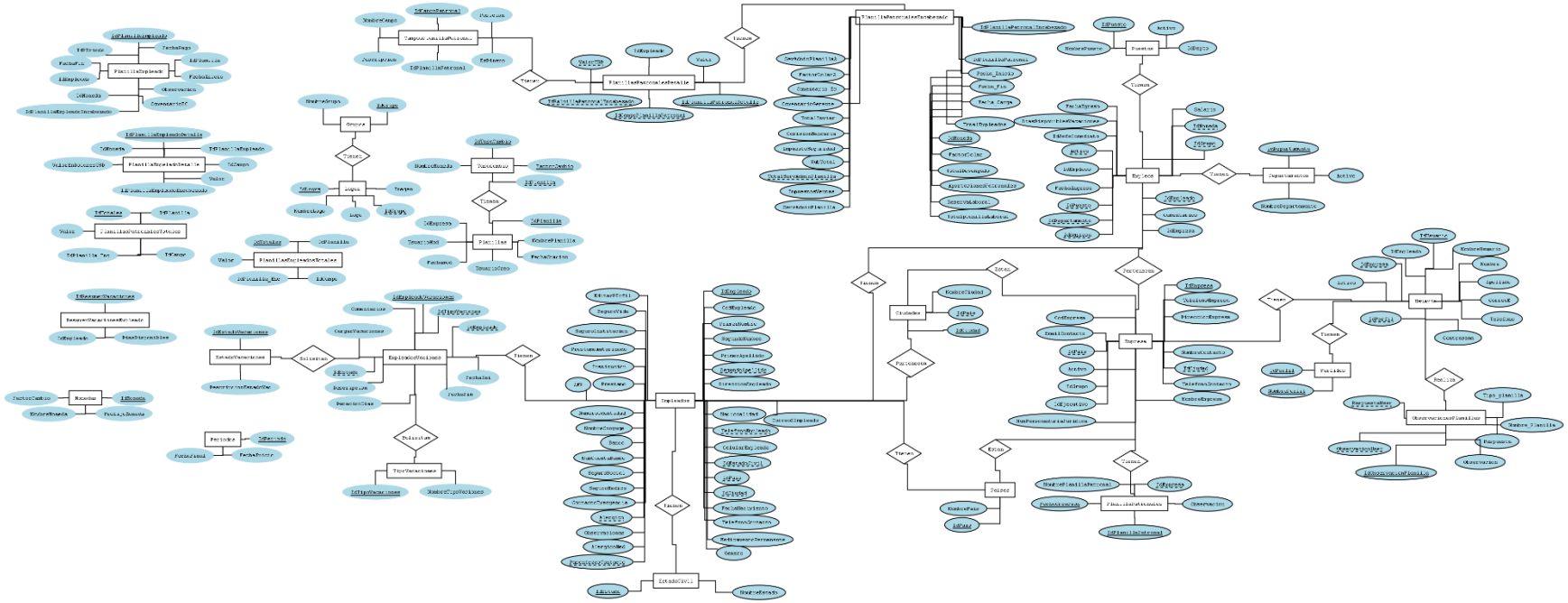


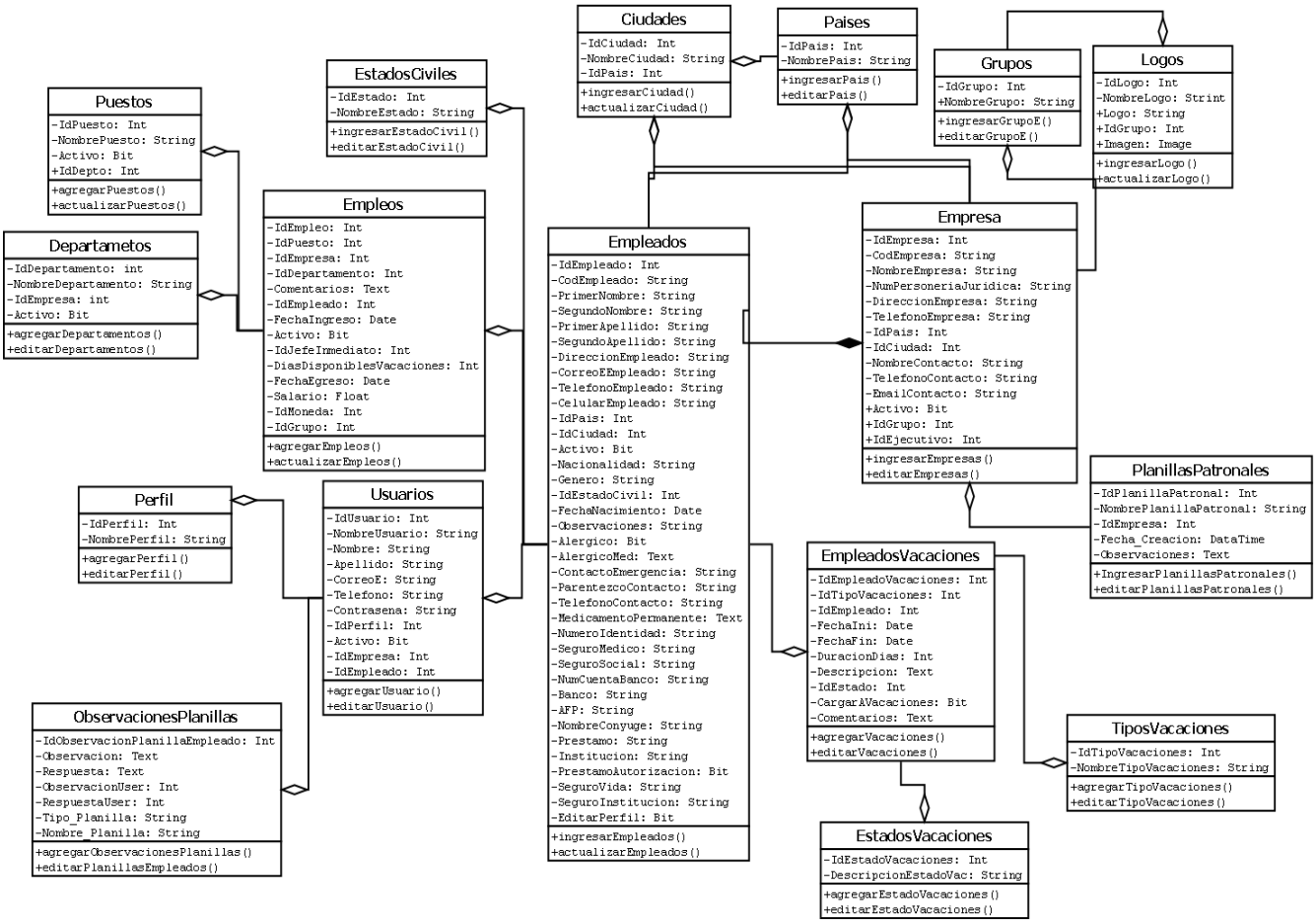
Diagrama de secuencia 11: Solicitud de Vacaciones



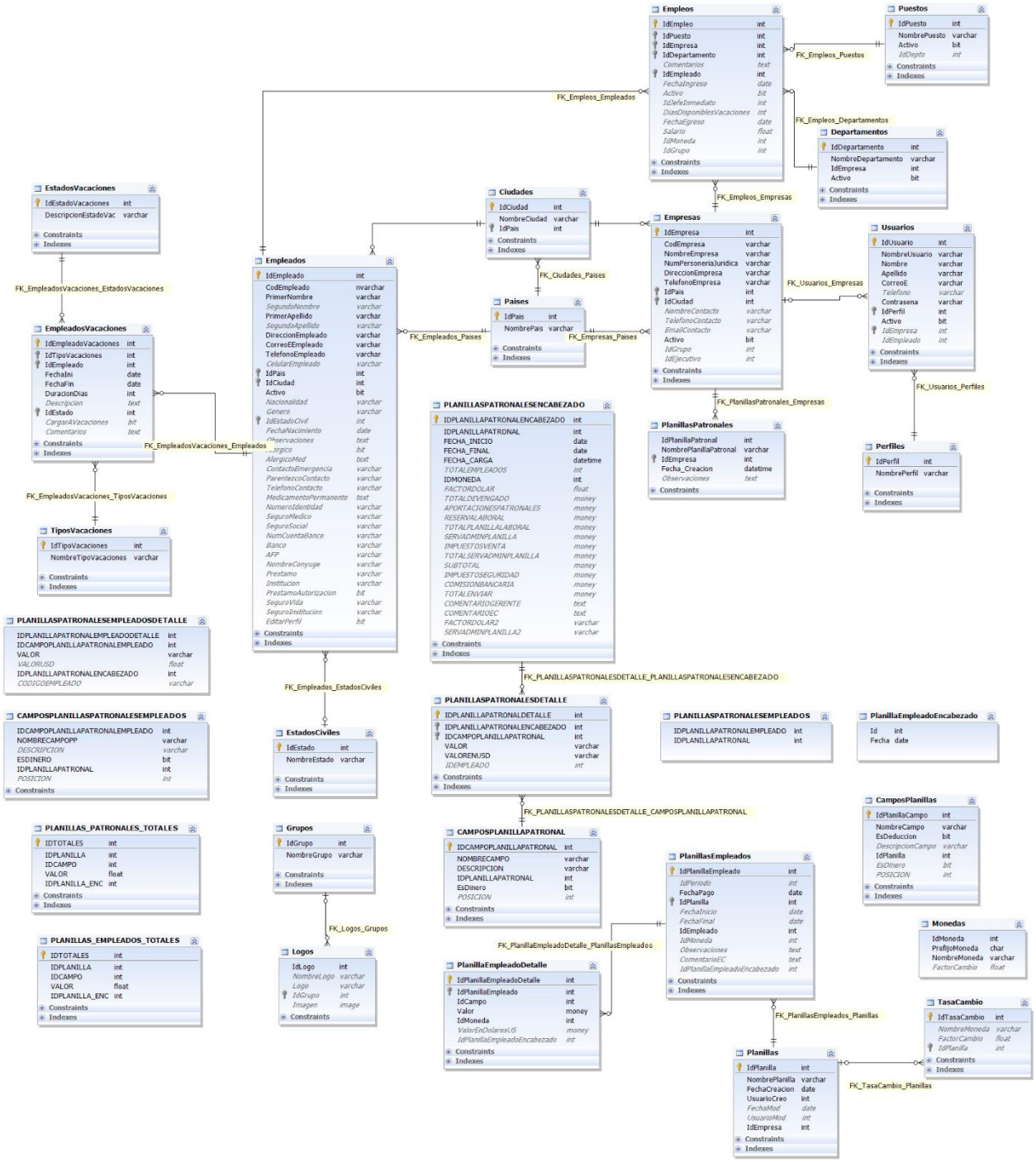
10. DIAGRAMA ENTIDAD RELACIÓN



11. DIAGRAMA DE CLASES



12. DIAGRAMA DE RELACIÓN



13. ANÁLISIS Y DISEÑO

13.1. Documentación de Tablas de Entidades

13.1.1 Estructura de tablas de la base de datos

13.1.1.1 Tabla Usuario

Almacena todos los datos de los Usuarios que se registrarán en la base de datos.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdUsuario	Identificador de usuario	PK	Int
NombreUsuario	Nombre de Usuario		varchar(100)
Nombre	Nombre completo del usuario		varchar(150)
Apellido	Apellido del usuario		varchar(150)
CorreoE	Correo Electrónico		varchar(200)
Telefono	Número de Teléfono		varchar(50)
Contrasena	Contraseña del usuario		varchar(50)
IdPerfil	Identificador del perfil del usuario		int
Activo	Verifica si está activo o inactivo el usuario		bit
IdEmpresa	Identifica a la empresa que pertenece el usuario		int
IdEmpleado	Identificador del empleado		int

13.1.1.2 Tabla Perfil

Almacena los nombres de los perfiles de cada empleado.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPerfil	Identifica el perfil	PK	Int
NombrePerfil	Nombre del perfil de usuario		varchar(100)

13.1.1.3 Tabla Empleados

Almacena todos los datos de los empleados que se registrarán en la base de datos.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEmpleado	Identificador del empleado	UNICODE	Int
CodEmpleado	Código que identifica al empleado	PK	varchar(50)
PrimerNombre	Primer nombre del empleado		varchar(100)
SegundoNombre	Segundo nombre del empleado		varchar(100)
PrimerApellido	Primer apellido del empleado		varchar(100)
SegundoApellido	Segundo apellido del empleado		varchar(100)
DireccionEmpleado	Dirección del empleado		varchar(500)
CorreoEEmpleado	Correo electrónico del empleado		varchar(300)

TelefonoEmpleado	Teléfono del empleado		varchar(50)
CelularEmpleado	Celular del empleado		varchar(50)
IdPais	Identificador del país que pertenece el empleado		Int
IdCiudad	Identificador de la ciudad que pertenece el empleado		Int
Activo	Identifica si el empleado está activo o inactivo		Bit
Nacionalidad	Nacionalidad del empleado		varchar(100)
Genero	Identifica el género que pertenece		varchar(30)
IdEstadoCivil	Identificador del estado civil		Int
FechaNacimiento	Fecha de nacimiento del empleado		Date
Observaciones	Observaciones del empleado		text
Alergico	Identifica si el empleado es alérgico		bit
AlergicoMed	Alérgico a algún medicamento		text
ContactoEmergencia	Contacto de emergencia		varchar(500)
ParentezcoContacto	Contacto de algún pariente		varchar(200)
TelefonoContacto	Contacto de teléfonos		varchar(50)
MedicamentoPermanente	Medicamento permanente del empleado		text
NumeroIdentidad	Numero de identidad		varchar(20)
SeguroMedico	Seguro medico		varchar(30)
SeguroSocial	Seguro social		varchar(30)

NumCuentaBanco	Número de cuenta del Banco		varchar(30)
Banco	Banco al que pertenece el empleado		varchar(100)
NombreConyuge	Nombre de cónyuge		varchar(200)
Prestamo	Prestamos		varchar(30)
Institucion	Es la institución que le realiza el prestamo		varchar(40)
PrestamoActorizacion	Identifica si el préstamo es autorizado o no		bit
SeguroVida	Almacena el número del seguro de vida		varchar(30)
SeguroInstitucion	Es la institución en la que está asegurado		varchar(50)
EditarPerfil	Identifica si puede editar su perfil o no		Bit

13.1.1.4 Tabla Estados civiles

Esta tabla solo tiene almacenados los estados civiles en la base de datos

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEstado	Identificador del estado civil	PK	int
NombreEstado	Nombre del estado civil		varchar(100)

13.1.1.5 Tabla Empresas

En esta tabla se almacena los datos de la empresa

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEmpresa	Identificador de la empresa	UNICODE	Int
CodEmpresa	Código de la empresa	PK	varchar(50)
NombreEmpresa	Nombre de la empresa		varchar(150)
NumPersoneriaJuridica	Número del personal Jurídico		varchar(50)
DireccionEmpresa	Dirección de la empresa		varchar(300)
TelefonoEmpresa	Teléfono de la empresa		varchar(50)
IdPais	Identificador del país que pertenece la empresa		Int
IdCiudad	Identificador de la ciudad que pertenece la empresa		Int
NombreContacto	Nombre de contacto de la empresa		varchar(200)
TelefonoContacto	Contacto Telefónico		varchar(150)
EmailContacto	Contacto de correo electrónico		varchar(300)
Activo	Identifica si la empresa está activa o no		Bit
IdGrupo	Identifica el grupo que pertenece la empresa		Int
IdEjecutivo	Identificador del ejecutivo de cuenta		Int

13.1.1.6 Tabla Países

Almacenara los países que se encuentran las empresas registradas

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPais	Identificador del país	Pk	int
NombrePais	Nombre del país		varchar(200)

13.1.1.7 Tabla Ciudades

Almacenara todas las ciudades correspondientes a cada país

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdCiudad	Identificador de la ciudad	PK	Int
NombreCiudad	Nombre de la ciudad		varchar(300)
IdPais	Identifica el país que pertenece la ciudad		Int

13.1.1.8 Tabla Grupo de Empresas

Almacenara los grupos de trabajos que pertenecen a la empresa

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdGrupo	Identificador del grupo de la empresa	PK	Int
NombreGrupo	Nombre del grupo de la empresa		varchar(200)

13.1.1.9 Tabla Logos

Almacenara los logos de la empresa y los de los grupos de trabajos

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdLogo	Identificador de logos de la empresa	PK	Int
NombreLogo	Nombre del logo de la empresa		varchar(100)
Logo	Logo de la empresa		varchar(500)
IdGrupo	Identificador del grupo de la empresa		Int
Imagen	Imagen del logo de la empresa		Image

13.1.1.10 Tabla Empleos

Esta tabla almacenara los datos relacionados a la empresa, grupo, puesto logos cundo se le asigna un empleo a un empleado.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEmpleo	Identificador del empleo		Int
IdPuesto	Identificador del puesto del empleo	PK	Int
IdEmpresa	Identificado de la empresa		Int
IdDepartamento	Identificador de departamento		Int
Comentarios	Comentarios del empleo		Text
IdEmpleado	Identificador del empleado		Int
FechaIngreso	Fecha de ingreso al puesto		Date

Activo	Identifica si está activo o inactivo		Bit
IdJefeInmediato	Identificador del jefe inmediato		Int
DiasDisponiblesVacaciones	Días disponibles de vacaciones		Int
FechaEgreso	Fecha de egreso al empleo		Date
Salario	Salario		Float
IdMoneda	Identifica el tipo de moneda		Int
IdGrupo	Identificador del grupo de la empresa		Int

13.1.1.11 Tabla Puesto

Almacena los puestos de trabajos que la empresa tiene

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPuesto	Identificador del puesto	PK	Int
NombrePuesto	Nombre del puesto		varchar(150)
Activo	Identifica si está activo o inactivo		Int
IdDepto	Identificador del departamento que pertenece		Int

13.1.1.12 Tabla Departamentos

La empresa está dividida en departamentos de trabajos lo cual esta tabla almacena esa información.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdDepartamento	Identificador del departamento	PK	Int
NombreDepartamento	Nombre del departamento		varchar(150)
IdEmpresa	Identificador de la empresa que pertenece		Int
Activo	Identifica si está activo o inactivo		Bit

13.1.1.13 Tabla Permisos Gerentes

Almacena los permisos que el gerente a asignado a los usuarios

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
idPermiso	Identificador de los permisos	PK	Int
IdEmpresa	Identifica la empresa a la que le dará los permisos		Int
IdUsuario	Identifica al usuario que le darán permiso		Int
Activo	Identifica si está activo o inactivo		Bit

13.1.1.14 Tabla Planillas

Almacena los registros de planillas cuando se crea

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPlanilla	Identificador de las planilla	PK	Int
NombrePlanilla	Nombre de la planilla		varchar(150)
FechaCreacion	Fecha de creación de la planilla		Date
UsuarioCreo	Usuario que creo la planilla		Int
FechaMod	Fecha que se modificó la planilla		date
UsuarioMod	Identifica al usuario que modifico la planilla		Int
IdEmpresa	Identifica la empresa que pertenece la planilla		Int

13.1.1.15 Tabla Campos Planillas

Esta tabla almacena los campos que contendrán las planillas de los empleados.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPlanillaCampo	Identifica el campo de la planilla		Int
NombreCampo	Nombre del campo de la planilla		varchar(150)
EsDeduccion	Identifica si es deducción o no		Bit
DescripcionCampo	Describe el campo de la planilla		varchar(500)
IdPlanilla	Identifica la planilla que pertenece		Int
EsDinero	Identifica si es dinero o no		Bit
POSICIÓN	Identifica la posición que se encuentra el campo de la planilla		Int

13.1.1.16 Tabla Planillas Empleados

Almacena la información de las planillas de los empleados

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPlanillaEmpleado	Identificador de la planilla empleado	PK	Int
IdPeriodo	Identificador del periodo de la planilla		Int
FechaPago	Fecha de pago		Date
IdPlanilla	Identificador de la planilla		Int

FechaInicio	Fecha de inicio		Date
FechaFinal	Fecha final		Date
IdEmpleado	Identificador del empleado al que pertenece la planilla		Int
IdMoneda	Identificador de la moneda que se pagó en la planilla		Int
Observaciones	Observaciones sobre las planillas		Text
ComentarioEC	Comentarios del ejecutivo de cuenta		Text
IdPlanillaEmpleadoEncabezado	Identificador de los encabezados de la planilla de los empleados		Int

13.1.1.17 Tabla Planillas Patronales

Almacena la información de las planillas patronales

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPlanillaPatronal	Identifica a la planilla patronal	PK	Int
NombrePlanillaPatronal	Nombre de la planilla patronal		varchar(100)
IdEmpresa	Identificador de la empresa a la que pertenece		Int
Fecha_Creacion	Fecha de creación de la planilla		Datetime
Observaciones	Observaciones		Text

13.1.1.18 Tabla Planillas empleados Detalles

Almacena los detalles de las planillas de los empleados

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdPlanillaEmpleadoDetalle	Identificador de los detalles de la planilla de los empleados	PK	Int
IdPlanillaEmpleado	Identificador de la planilla de los empleados		Int
IdCampo	Identificador del campo de la panilla		Int
Valor	Valor de la moneda		Money
IdMoneda	Identificador de la moneda		Int
ValorEnDolaresUS	Valor en dólares		Money
IdPlanillaEmpleadoEncabezado	Identificador del encabezado de la planilla de los empleados		Int

13.1.1.19 Tabla Planillas Patronales Detalles

Almacena los detalles de las planillas patronales de las empresas

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IDPLANILLAPATRONALDETALLE	Identificador de la planilla patronales detalles	PK	Int
IDPLANILLAPATRONALENCABEZADO	Identifica los encabezado de las planillas patronales		Int
IDCAMPOPLANILLAPATRONAL	Identifica los campos de las planillas patronales		Int
VALOR	Valor		varchar(50)
VALORENUSD	Valor en dólares		varchar(50)
IDEMPLEADO	Identifica al empleado		Int

13.1.1.20 Tabla Observaciones de Planillas

Almacenara las observaciones de las planillas

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdObservacionPlanillaEmpleado	Identificador	PK	Int
Observacion	Observaciones		Text
Respuesta	Respuestas		Text
ObservacionUser	Observación del usuario		Int
RespuestaUser	Respuesta del usuario		Int
Tipo_Planilla	Tipo de planilla		varchar(100)
Nombre_Planilla	Nombre de la planilla		varchar(120)

13.1.1.21 Tabla Planilla Empleado Total

Almacena el total de todas las planillas de los empleados

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IDTOTALES	Identificador de la tabla planilla empleado total	PK	Int
IDPLANILLA	Identificador de la planilla		Int
IDCAMPO	Identificador del campo de la planillas		Int
VALOR	Valor		Float
IDPLANILLA_ENC	Identificador del encabezado de la planilla empleado		Int

13.1.1.22 Tabla Planilla Patronales Total

Almacena el total de todas las planillas patronales

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IDTOTALES	Identificador de la tabla planilla patronales total	PK	Int
IDPLANILLA	Identificador de la planilla		Int
IDCAMPO	Identificador del campo de la planillas		Int
VALOR	Valor		Float
IDPLANILLA_ENC	Identificador del encabezado de la planilla patronal		Int

13.1.1.23 Tabla Tasa de Cambio

Almacena el tipo de cambio que trabaja la empresa a la hora de pagar a los empleados.

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdTasaCambio	Identificador de la tasa de cambio	PK	Int
NombreMoneda	Nombre de la moneda		varchar(100)
FactorCambio	Factor de cambio de la moneda		Float
IdPlanilla	Identificador de la planilla		Int

13.1.1.24 Tabla Monedas

Almacena la moneda con la que trabaja la empresa

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdMoneda	Identificador de la moneda	PK	Int
PrefijoMoneda	Prefijo de la moneda		Char(4)
NombreMoneda	Nombre de la moneda		varchar(100)
FactorCambio	Factor cambio de la moneda		Float

13.1.1.25 Tabla Tipos de Vacaciones

Almacena los tipos de vacaciones que tiene la empresa

NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdTipoVacaciones	Identificador del tipo de vacaciones	PK	Int
NombreTipoVacaciones	Nombre del tipo de vacaciones		varchar(150)

13.1.1.26 Tabla Empleados Vacaciones

Almacena la información de los empleados con vacaciones

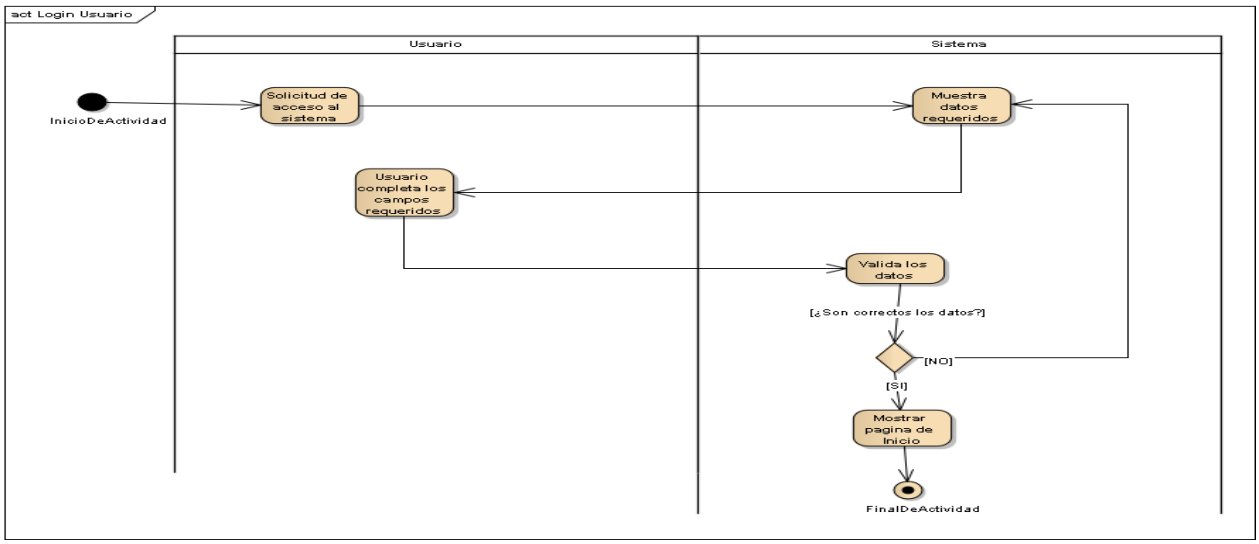
NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEmpleadoVacaciones	Identifica a la tabla empleado vacaciones	PK	Int
IdTipoVacaciones	Identifica el tipo de vacaciones		Int
IdEmpleado	Identifica al empleado que realiza la vacación		Int
FechaIni	Fecha de inicio		Date
FechaFin	Fecha de fin		Date
DuracionDias	Días de duración de las vacaciones		Int
Descripcion	Descripción de la vacación		Text
IdEstado	Identifica el estado de la vacación		Int
CargarAVacaciones	Carga las vacaciones		Bit
Comentarios	Comentarios de la vacaciones		Text

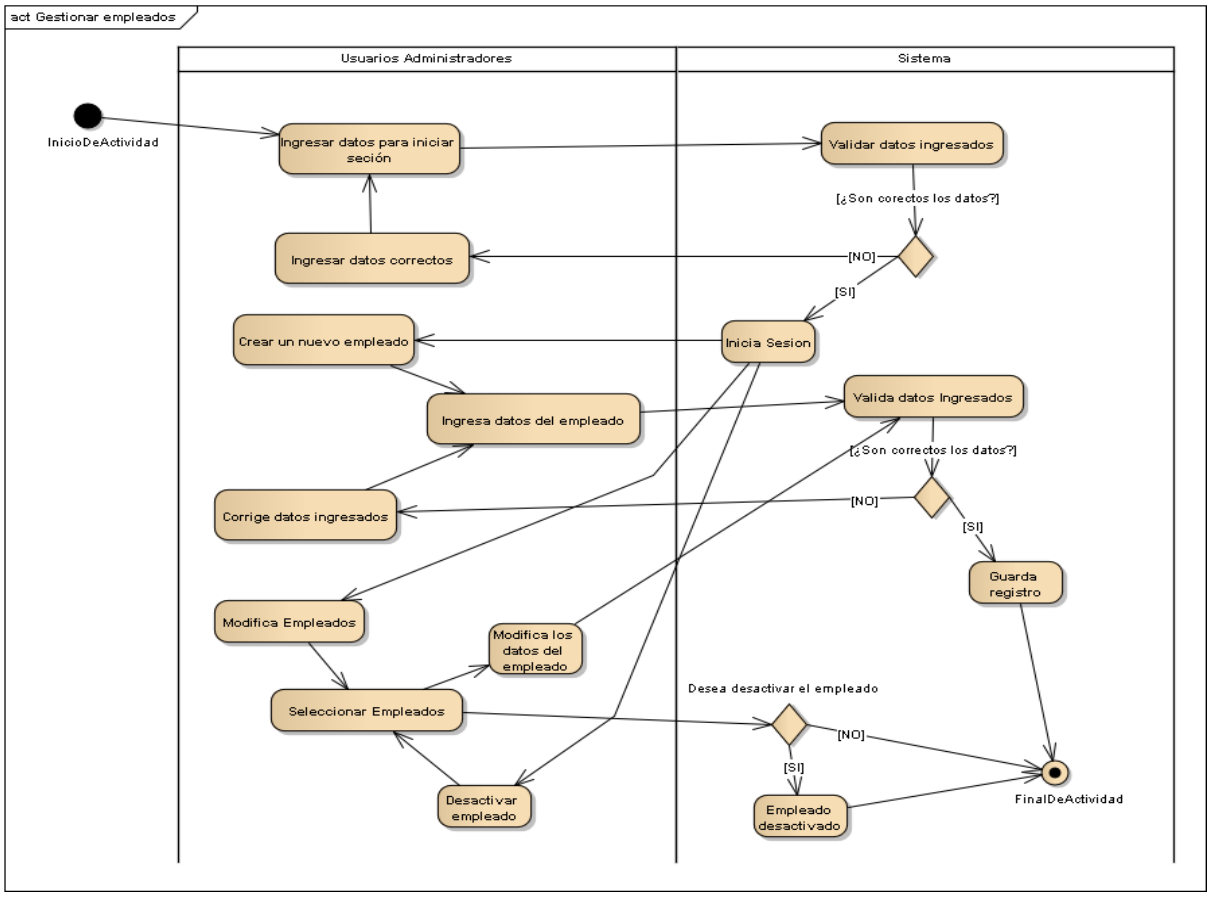
13.1.1.27 Tabla Estados Vacaciones

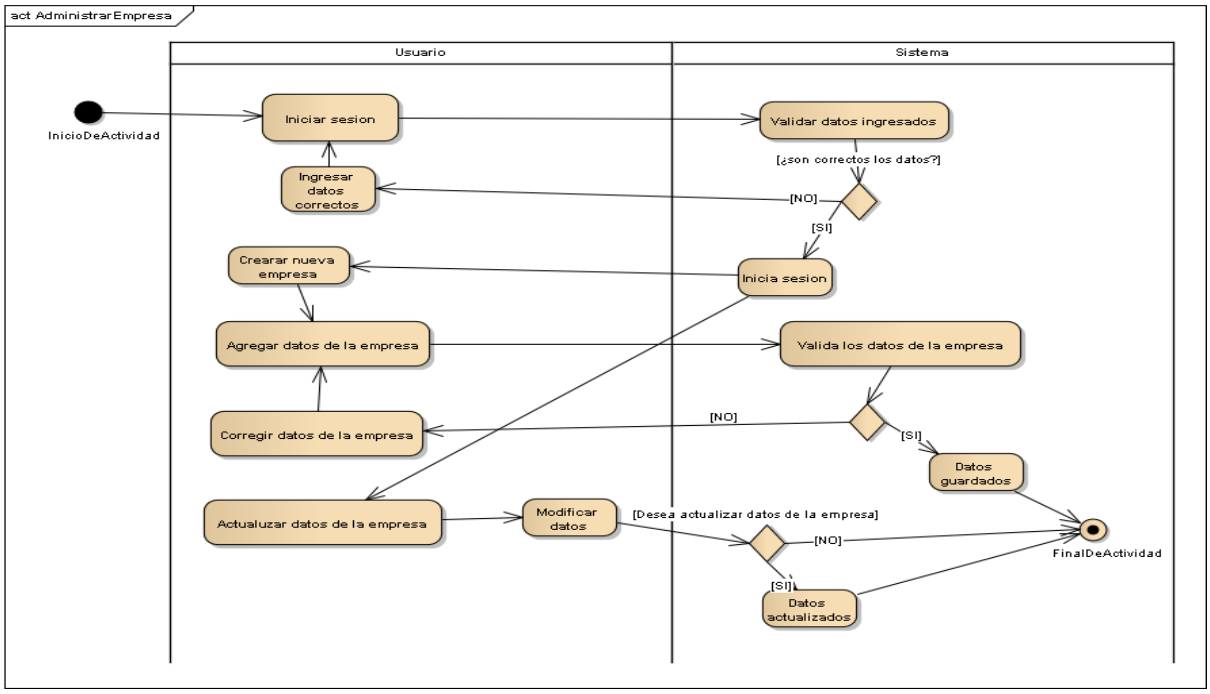
Almacena los estados de las vacaciones

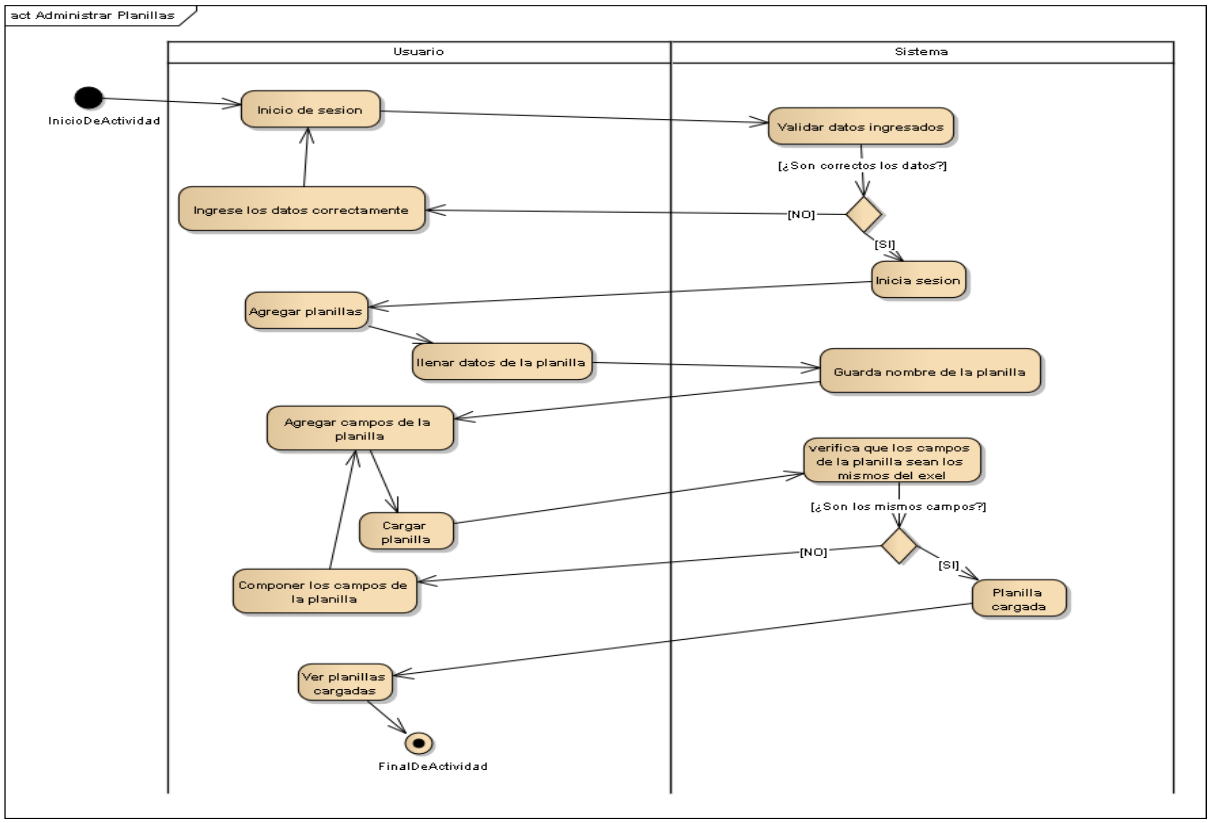
NOMBRE CAMPO	DESCRIPCIÓN	LLAVE	TIPO DE DATO
IdEstadoVacaciones	Identifica el estado de la vacación	PK	Int
DescripcionEstadoVac	Descripción del estado de las vacaciones		varchar(150)

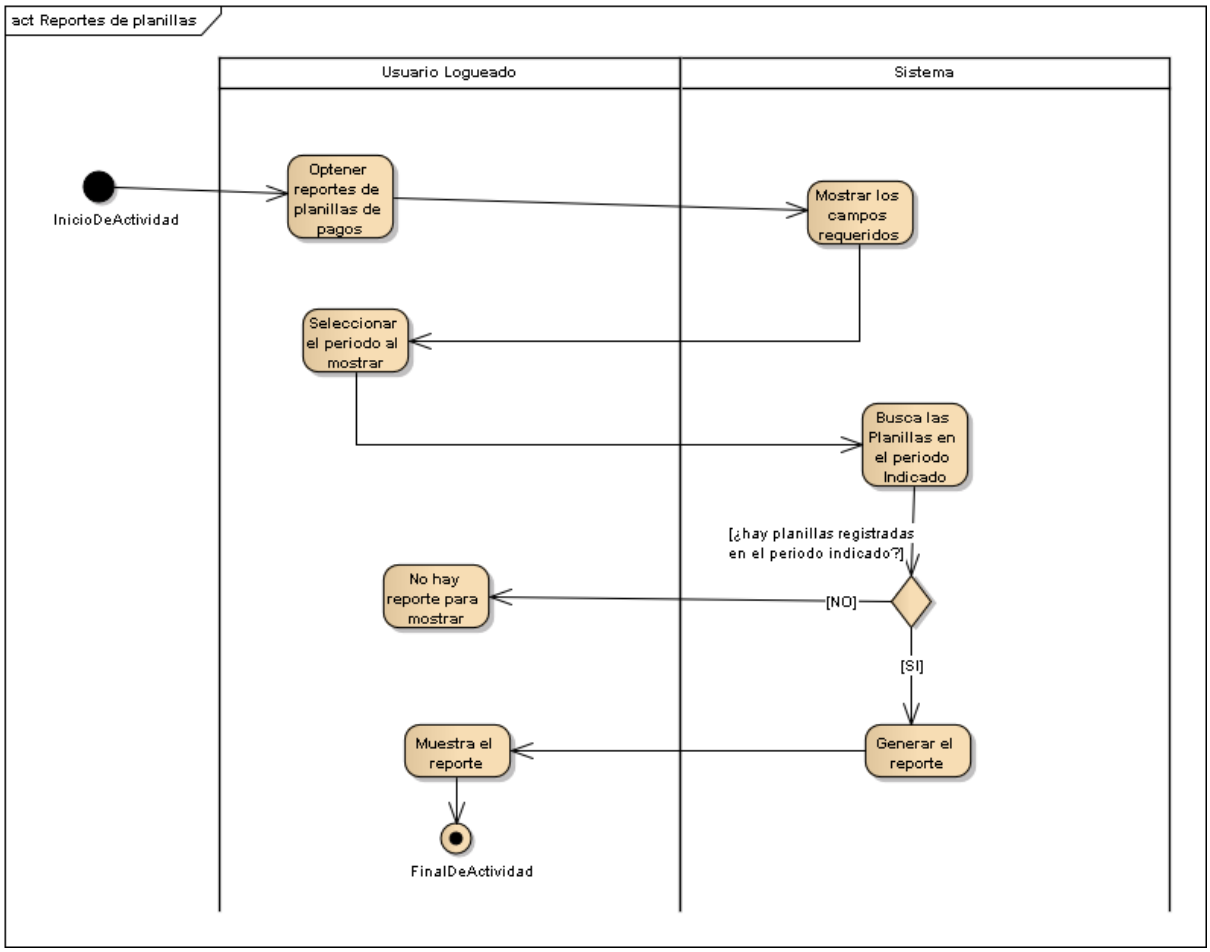
13.2. Diagrama de Actividades



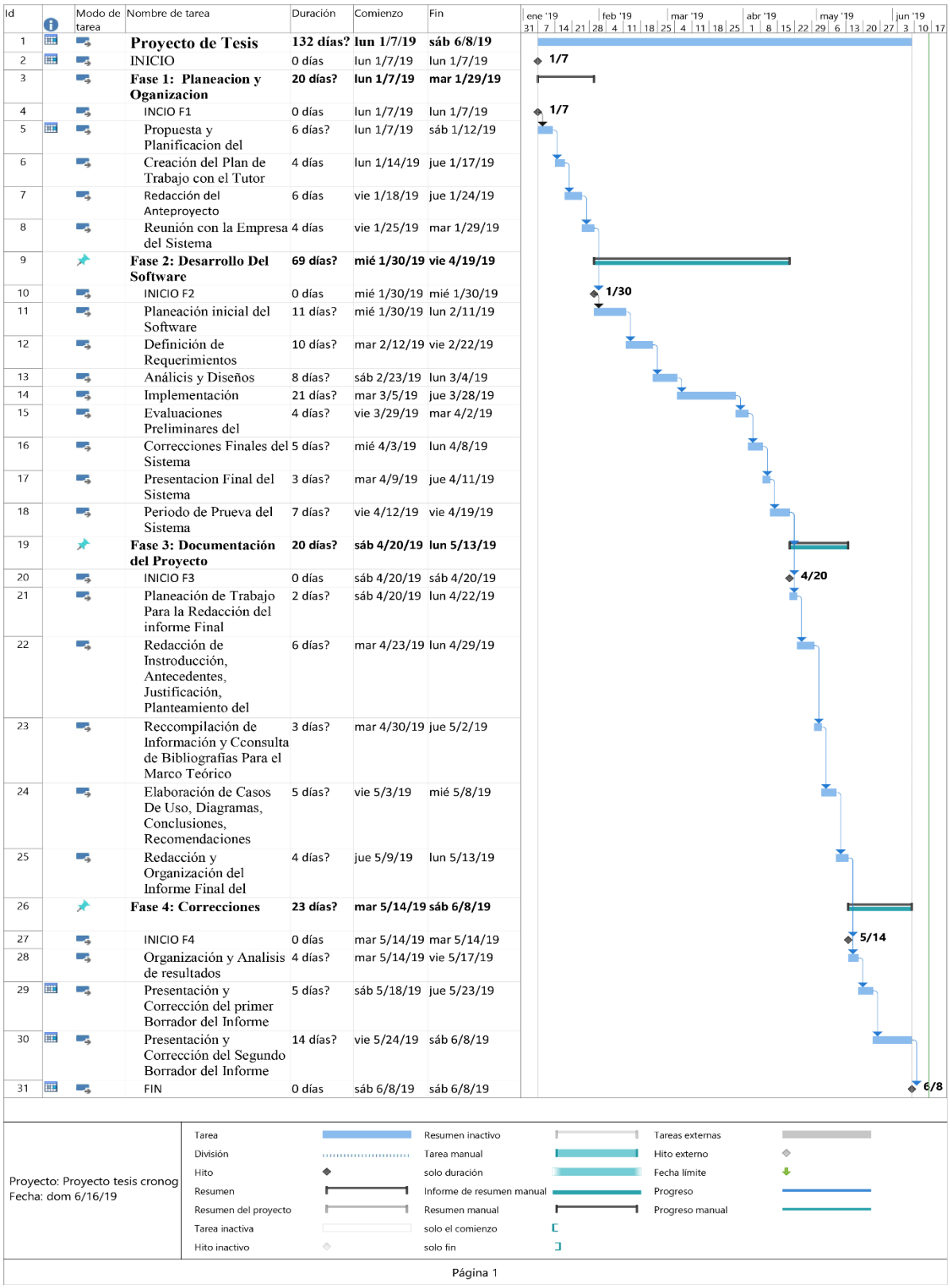



act AdministrarEmpresa
Usuario
Sistema
InicioDeActividad
Iniciar sesion
Validar datos ingresados
[¿son correctos los datos?]
Ingresar datos correctos
[NO]
[SI]
Crearar nueva empresa
Inicia sesion
Agregar datos de la empresa
Valida los datos de la empresa
Corregir datos de la empresa
[NO]
[SI]
Datos guardados
Actualuzar datos de la empresa
Modificar datos
[Desea actualizar datos de la empresa]
[NO]
[SI]
Datos actualizados
FinalDeActividad


act Administrar Planillas
Usuario
Sistema
InicioDeActividad
Inicio de sesion
Validar datos ingresados
[¿Son correctos los datos?]
Ingrese los datos correctamente
[NO]
[SI]
Inicia sesion
Agregar planillas
llenar datos de la planilla
Guarda nombre de la planilla
Agregar campos de la planilla
verifica que los campos de la planilla sean los mismos del exel
[¿Son los mismos campos?]
Cargar planilla
[NO]
[SI]
Componer los campos de la planilla
Planilla cargada
Ver planillas cargadas
FinalDeActividad


act Reportes de planillas
Usuario Logueado
Sistema
InicioDeActividad
Optener reportes de planillas de pagos
Mostrar los campos requeridos
Seleccionar el periodo al mostrar
Busca las Planillas en el periodo Indicado
[¿hay planillas registradas en el periodo indicado?]
[NO]
No hay reporte para mostrar
[SI]
Generar el reporte
Muestra el reporte
FinalDeActividad

14. CRONOGRAMA



Id	Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin
1		**Proyecto de Tesis**	**132 días?**	**lun 1/7/19**	**sáb 6/8/19**
2		INICIO	0 días	lun 1/7/19	lun 1/7/19
3		**Fase 1: Planeacion y Oganizacion**	**20 días?**	**lun 1/7/19**	**mar 1/29/19**
4		INCIO F1	0 días	lun 1/7/19	lun 1/7/19
5		Propuesta y Planificacion del	6 días?	lun 1/7/19	sáb 1/12/19
6		Creación del Plan de Trabajo con el Tutor	4 días	lun 1/14/19	jue 1/17/19
7		Redacción del Anteproyecto	6 días	vie 1/18/19	jue 1/24/19
8		Reunión con la Empresa del Sistema	4 días	vie 1/25/19	mar 1/29/19
9		**Fase 2: Desarrollo Del Software**	**69 días?**	**mié 1/30/19**	**vie 4/19/19**
10		INICIO F2	0 días	mié 1/30/19	mié 1/30/19
11		Planeación inicial del Software	11 días?	mié 1/30/19	lun 2/11/19
12		Definición de Requerimientos	10 días?	mar 2/12/19	vie 2/22/19
13		Análicis y Diseños	8 días?	sáb 2/23/19	lun 3/4/19
14		Implementación	21 días?	mar 3/5/19	jue 3/28/19
15		Evaluaciones Preliminares del	4 días?	vie 3/29/19	mar 4/2/19
16		Correcciones Finales del Sistema	5 días?	mié 4/3/19	lun 4/8/19
17		Presentacion Final del Sistema	3 días?	mar 4/9/19	jue 4/11/19
18		Periodo de Prueva del Sistema	7 días?	vie 4/12/19	vie 4/19/19
19		**Fase 3: Documentación del Proyecto**	**20 días?**	**sáb 4/20/19**	**lun 5/13/19**
20		INICIO F3	0 días	sáb 4/20/19	sáb 4/20/19
21		Planeación de Trabajo Para la Redacción del informe Final	2 días?	sáb 4/20/19	lun 4/22/19
22		Redacción de Instroducción, Antecedentes, Justificación, Planteamiento del	6 días?	mar 4/23/19	lun 4/29/19
23		Reccompilación de Información y Cconsulta de Bibliografías Para el Marco Teórico	3 días?	mar 4/30/19	jue 5/2/19
24		Elaboración de Casos De Uso, Diagramas, Conclusiones, Recomendaciones	5 días?	vie 5/3/19	mié 5/8/19
25		Redacción y Organización del Informe Final del	4 días?	jue 5/9/19	lun 5/13/19
26		**Fase 4: Correcciones**	**23 días?**	**mar 5/14/19**	**sáb 6/8/19**
27		INICIO F4	0 días	mar 5/14/19	mar 5/14/19
28		Organización y Analisis de resultados	4 días?	mar 5/14/19	vie 5/17/19
29		Presentación y Corrección del primer Borrador del Informe	5 días?	sáb 5/18/19	jue 5/23/19
30		Presentación y Corrección del Segundo Borrador del Informe	14 días?	vie 5/24/19	sáb 6/8/19
31		FIN	0 días	sáb 6/8/19	sáb 6/8/19

Proyecto: Proyecto tesis cronog
Fecha: dom 6/16/19

Tarea	Resumen inactivo	Tareas externas	
División	Tarea manual	Hito externo	
Hito	solo duración	Fecha límite	
Resumen	Informe de resumen manual	Progreso	
Resumen del proyecto	Resumen manual	Progreso manual	
Tarea inactiva	solo el comienzo		
Hito inactivo	solo fin		

Página 1

15. CONCLUSIONES

Al culminar nuestro trabajo monográfico se llegó a las siguientes conclusiones:

Hemos logrado cumplir con cada uno de los objetivos propuestos, dando como respuesta el desarrollo de la aplicación: ***"sistema web en ASP.net para la gestión de recursos humanos basada en nóminas a la empresa ACAPROF, S.A ubicada en Matagalpa Nicaragua",*** basado en los conocimientos adquiridos sobre Programación y Base de Datos, al mismo tiempo, aplicamos técnicas o metodologías de Ingeniería del Software que involucran una serie de Herramientas, Métodos y Procedimientos que nos han permitido la implementación y desarrollo de dicho Sistema.

La aplicación realizada cubre los procesos que la empresa brinda a los empleados, garantizando la disminución en el tiempo de entrega de Informe a los empleados. Cabe destacar que la aplicación cuenta con un Sistema de Control de Usuario, para dar de alta y baja a las personas que tendrán acceso al Sistema, de esta manera, cada Usuario registrado accederá únicamente a la sección del Sistema que se le ha asignado.

Concluimos que es factible la operatividad del sistema una vez instalado, ya que fue diseñado para ser fácil de usar, y presenta un ambiente agradable para el usuario.

El hecho de que la aplicación le pertenezca a la institución le da la autorización de arreglar posibles averías de manera inmediata, además de realizar futuras modificaciones a medida que el sistema crezca.

Por consiguiente, nuestra propuesta satisfizo los objetivos planteados, logró implementar la mayoría de las funcionalidades requeridas para un sistema de planillas o nóminas de la empresa ACAPROF S.A.

16. RECOMENDACIONES

En base a los resultados obtenidos de nuestro trabajo y la consideración de que este software se llegue a implementar hacemos las siguientes recomendaciones:

16.1 Recomendaciones de uso o implementación

- La Creación de un manual para la manipulación del software y hardware.
- Capacitar a usuarios sobre cómo manejar la aplicación.
- Sensibilizar a los usuarios del uso y la integración de nuevas tecnologías a los procesos de labores diarias.

16.2 Mejoras o trabajos futuros.

- Desarrollar una aplicación móvil que pueda hacer lo mismo que la aplicación web.
- Agregar nuevas funcionalidades para que el sistema interactúe en perfectas condiciones con las aplicaciones que ya tiene la empresa en torno a las nóminas.
  - Viendo las necesidades de la empresa también se puede automatizar otras actividades en otras áreas como es la postulación de nuevos empleados por vía web, obtener sus certificados de trabajo automáticamente vía web sin necesidad de estar recurriendo constantemente a la oficina de recursos humanos.
  - Desarrollar una extensión que le permita llevar la contabilidad a las empresas.

17. BIBLIOGRAFÍA

CEBALLOS, F. J. (2007). Enciclopedia de Microsoft Visual C#. 2da. España.: Editorial RA-MA. Madrid.

Arias, A., & Durango, A. (2007). Curso de programación y análisis de software (3era ed.). Campus Academy.

Infografías

https://definicion.de/encriptar/. (s.f.).

https://en.wikipedia.org/wiki/DevExpress. (s.f.).

https://es.wikipedia.org/wiki/ASP.NET. (s.f.).

https://es.wikipedia.org/wiki/Bootstrap_(framework). (s.f.).

https://es.wikipedia.org/wiki/MD5. (s.f.).

https://es.wikipedia.org/wiki/Visual_Basic. (s.f.).

https://techlandia.com/funciona-aes-info_215975/. (s.f.).

https://getbootstrap.com/docs/4.3/getting-started/introduction/

https://documentation.devexpress.com/GeneralInformation/2216/Gereral-Information.

18. ANEXOS

Manual de Usuario del Sistema de Nomina “ACAPROF”

Para usuarios con diferentes roles

18.1 Introducción

Sistema de Nomina ACAPROF, es una herramienta web desarrollada bajo la plataforma de .NET con el propósito de llevar un control digitalizado, donde se puedan gestionar y administrar las planillas de los empleados, se podrá llevar un historial de los datos de cada empleado, así como sus boletas de pagos, gestionar sus vacaciones y controlar quienes están de forma activa en la empresa.

18.1.1 ¿Cuál es el objetivo del Manual de Usuarios?

El objetivo que se persigue con el presente manual, es dar a conocer a los usuarios las características y el modo de funcionamiento del sistema de Nomina ACAPROF.

18.2 Requerimiento Básico.

Para la utilización de la aplicación informática “Sistema de Nomina ACAPROF” se debe tener instalado localmente.

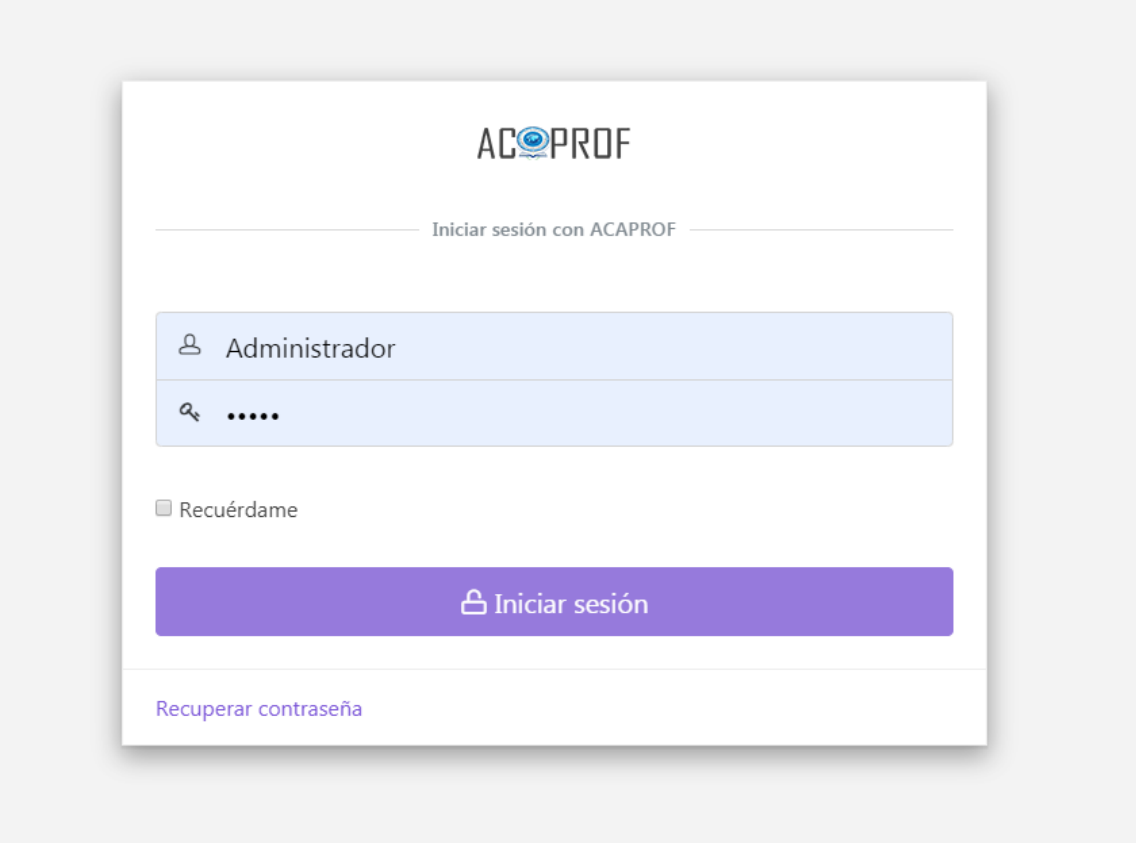
1. Transact Sql Server.
2. Configurado y montado el sitio en servidor IIS.
3. Un navegador cualquiera (preferiblemente Internet Explorer 7.0 o Mozilla).

El sistema es una aplicación web, lo que permite que una vez instalado en una máquina, se pueda acceder desde cualquier otra que esté conectada a la misma red.

18.3 Acceso a la aplicación

Para ingresar al sistema desde cualquier dispositivo o computadora, se debe acceder a la intranet de la oficina central de ACAPROF, mediante la siguiente URL: http://192.168.1.19:8090/, automáticamente se visualizará la ventana de la ***Figura 1,*** esta es la pantalla de autenticación del sistema y permite que solo puedan acceder los usuarios establecidos. Si es usuario del sistema, se deberá registrar en los campos mostrados en pantalla (***Usuario y Contraseña***), luego se debe pulsar el botón ***Iniciar sesión,*** de esta forma se introducirá en el sistema, tanto el destino como los privilegios del rol que ocupe dentro de la aplicación.

Figura 1. Pantalla de autenticación



18.4 Roles.

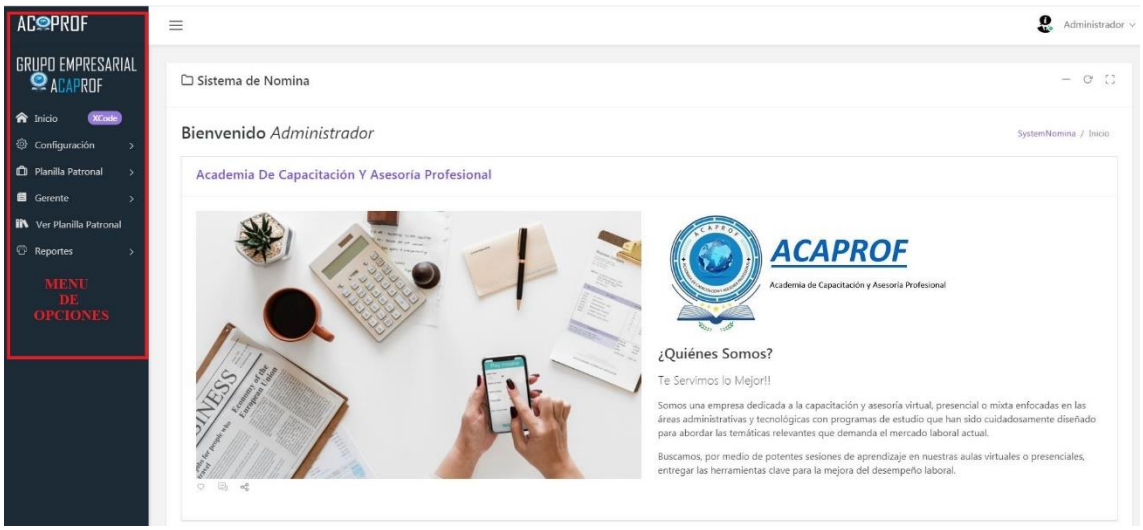
El rol es la clasificación que tienen los usuarios dentro del sistema, de manera que así se controla el manejo de la información por cada usuario dando determinados privilegios a los mismos según el rol.

- **Administrador:** Posee todos los privilegios dentro del sistema, generalmente son sus creadores y/o personas capaces de manejar toda su información y en caso de que fuera necesario la modificación del mismo.
- **Ejecutivo de cuenta:** Lo poseen los usuarios encargados de revisar y gestionar la información de los empleados que corresponden a la empresa asignada a ellos.
- **Gerente:** Lo posee únicamente los usuarios que están a cargo de una empresa, es al que se le denomina cliente, podrá visualizar la información de sus empleados, aprobar planillas, generar reportes, únicamente de la empresa el cual él está a cargo.
- **Empleado:** Serán los empleados que están en la empresa, solo podrán ver su perfil, boletas de pagos, deducciones y solicitar vacaciones, este usuario solo tendrá permiso de lectura en el sistema.

18.5 Pantalla de bienvenida

Después de que el usuario se autentica se cargara la página de inicio o bienvenida, se mostrará un menú con diferentes opciones que al ser seleccionadas nos cargará la acción que corresponde a la selección.

Figura 2. Pantalla de inicio



18.6 Perfil administrador

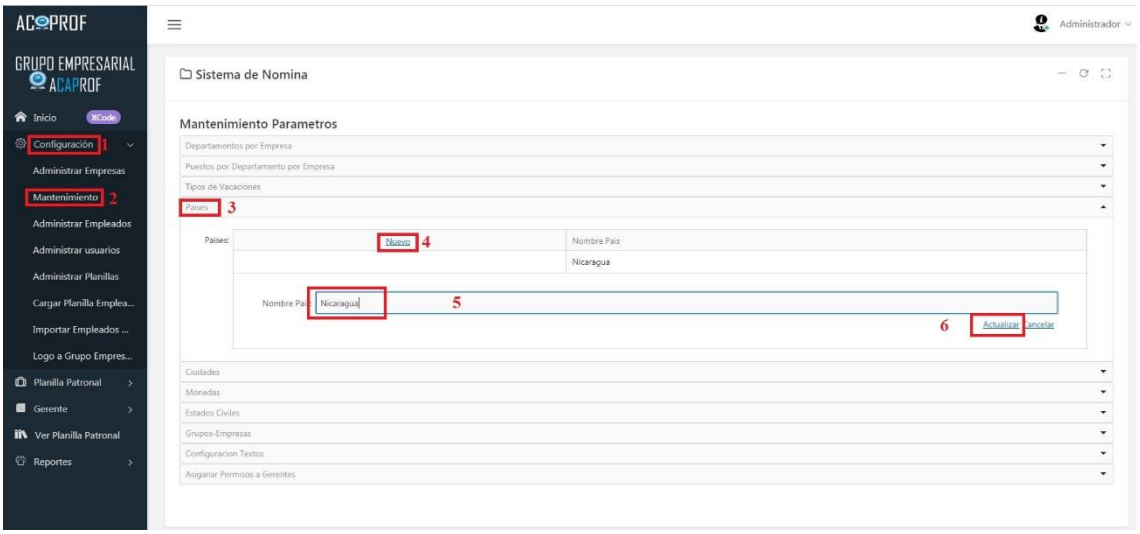
Este usuario será el encargado de configurar el sistema para que los demás usuarios con diferentes perfiles o roles pueda acceder al sistema y realizar las tareas asignadas, el administrador tendrá el acceso de manipular y gestionar todas las funciones que cualquier otro usuario pueda hacer, pero su principal función será configurar el sistema y si se requiere realizar modificaciones.

18.6.1 Configuración básica requerida

18.6.1.1 Agregar País

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar el menú mantenimiento, abrirá una página donde se podrán agregar diferentes datos que son requeridos por el sistema.
3. Seleccionar país, se desplegará una tabla con la lista de países si existen.
4. Seleccionar nuevo, muestra un formulario.
5. Se rellena el formulario con el nombre del país.
6. Se da clic en actualizar y automáticamente queda registrado el país.

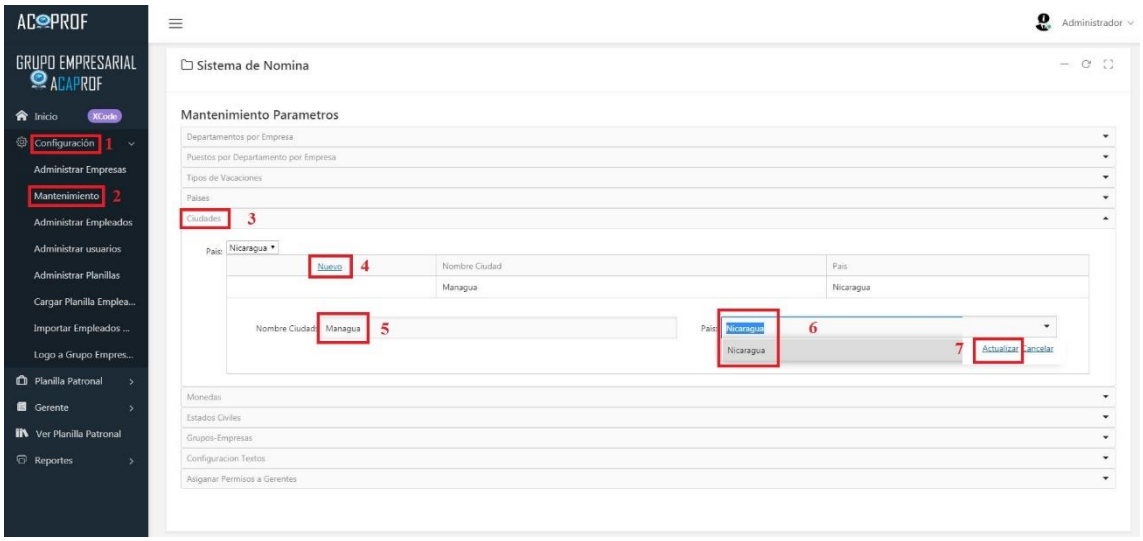
Figura 3. Agregar país



18.6.1.2 Agregar ciudad

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar el menú mantenimiento, abrirá una página donde se podrán agregar diferentes datos que son requeridos por el sistema.
3. Seleccionar ciudades, se desplegará una tabla con la lista de las ciudades si existen.
4. Seleccionar nuevo, muestra un formulario.
5. Se rellena el formulario con el nombre de la ciudad.
6. Se selecciona el país al que pertenece la ciudad en una lista desplegable.
7. Se da clic en actualizar y automáticamente queda registrado la ciudad.

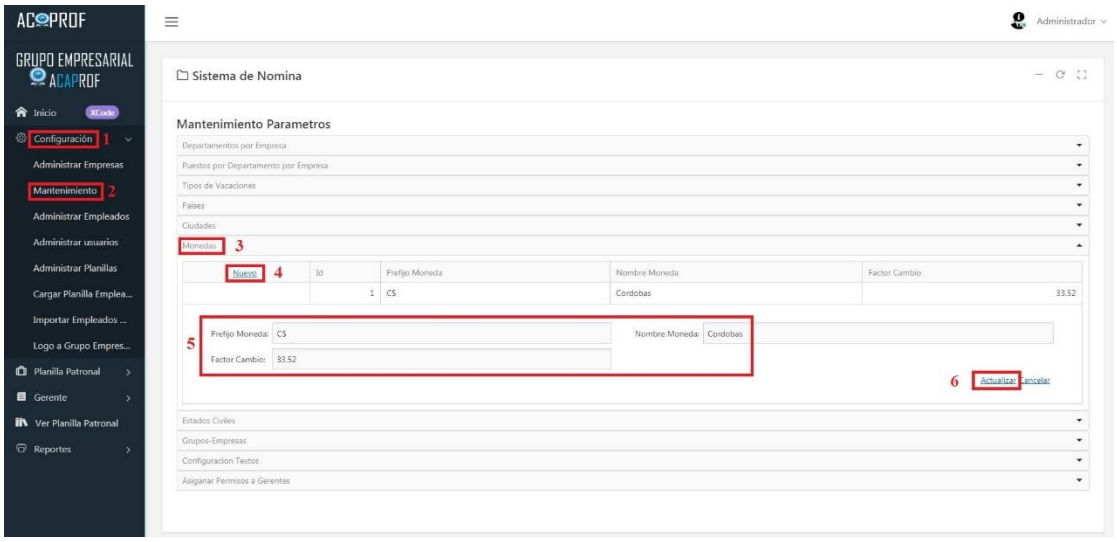
Figura 4. Agregar Ciudad



18.6.1.3 Agregar moneda

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar el menú mantenimiento, abrirá una página donde se podrán agregar diferentes datos que son requeridos por el sistema.
3. Seleccionar monedas, se desplegará una tabla con la lista de las monedas si existen.
4. Seleccionar nuevo, muestra un formulario.
5. Llenar el formulario con Prefijo, nombre y factor de cambio.
6. Se da clic en actualizar y automáticamente queda registrada la moneda.

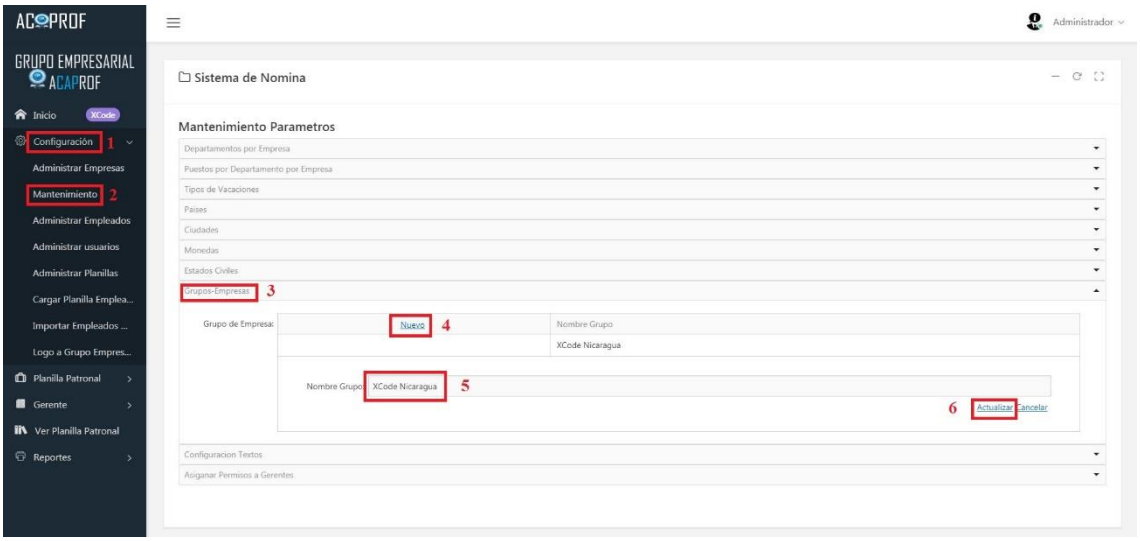
Figura 5. Agregar moneda



18.6.1.4 Agregar grupo empresarial

1. Seleccionar el menú configuración, desplegará el menú de opciones.
2. Seleccionar el menú mantenimiento, abrirá una página donde se podrán agregar diferentes datos que son requeridos por el sistema.
3. Seleccionar Grupos-Empresas, se desplegará una tabla con la lista de los grupos de las empresas si existen.
4. Seleccionar nuevo, muestra un formulario.
5. Llenar el formulario con el nombre del grupo.
6. Se da clic en actualizar y automáticamente queda registrado el nombre del grupo empresarial.

Figura 6. Agregar grupo empresarial.



18.6.2 Agregar empresa

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar *administrar empresas*, se desplegará una tabla con la lista de las empresas si existen.
3. Dar clic en el botón *agregar empresa*
4. Llenar el formulario con la información solicitada de la empresa.
5. Dar clic en el botón guardar.

Figura 7. Agregar empresa

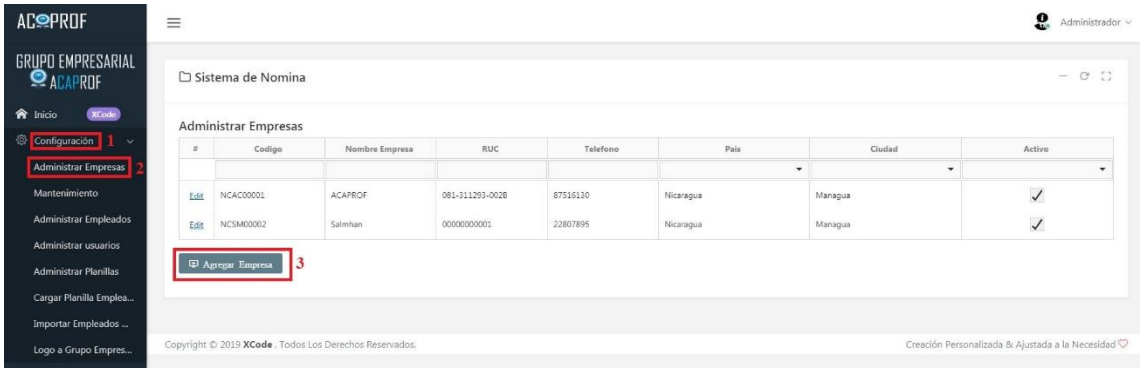
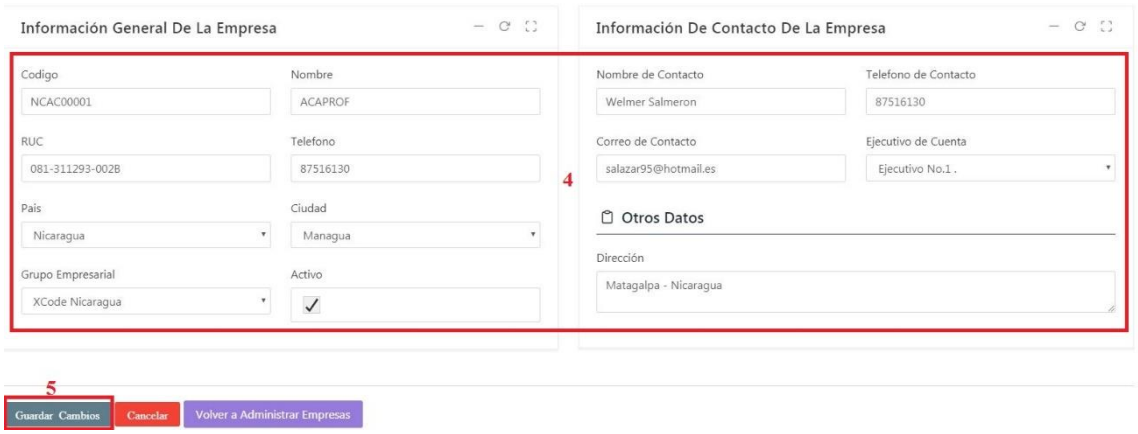


Figura 7.1 Formulario agregar empresa



18.6.3 Agregar perfil de empleado

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar *administrar empleados*, se desplegará una tabla con la lista de los empleados si existen.
3. Dar clic en el botón *agregar empleado.*
4. Llenar el formulario con la información solicitada del empleado.
5. Dar clic en el botón guardar.

Figura 8. Agregar empleado

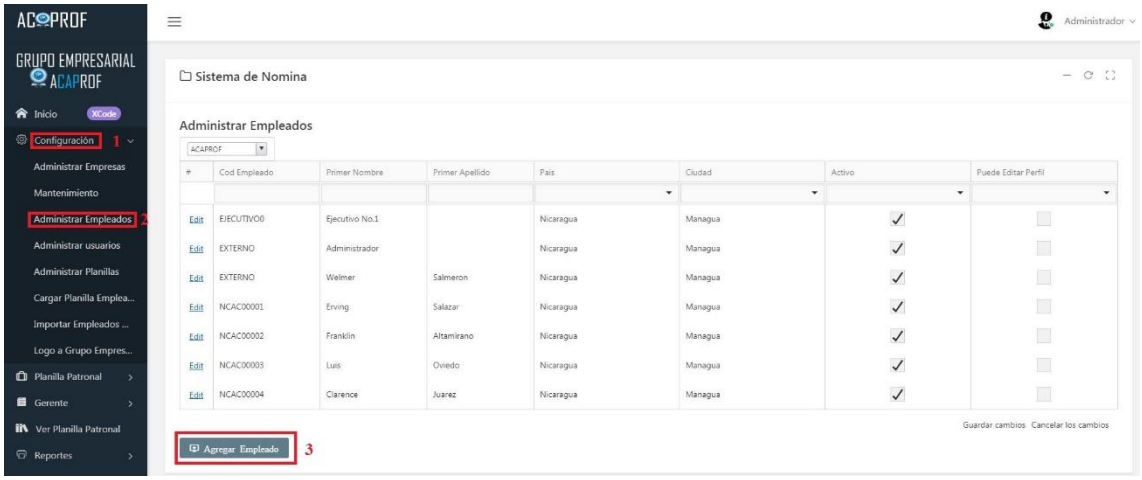
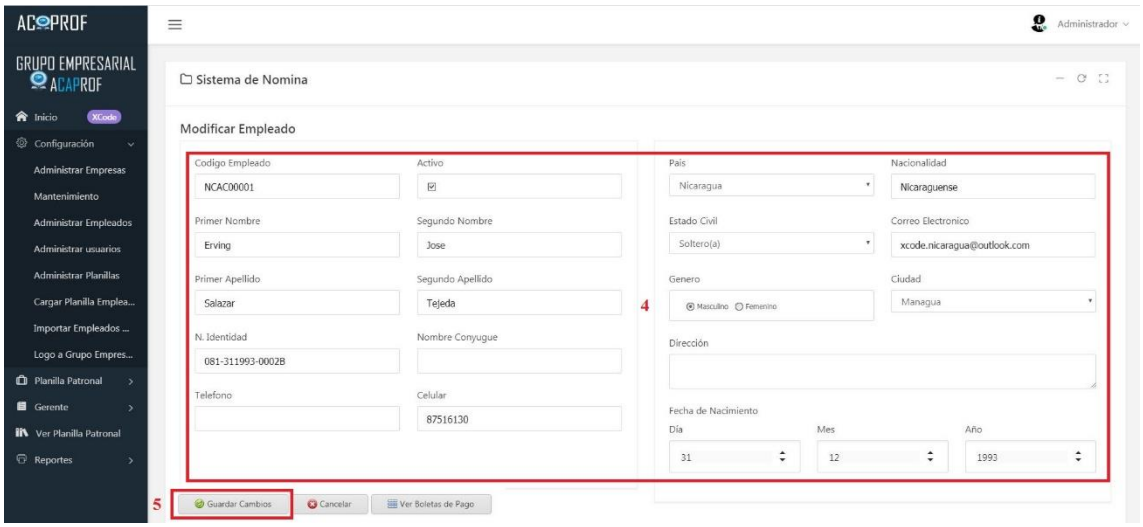


Figura 8.1. Formulario agregar empleado.



18.6.4 Agregar usuario

1. Seleccionar el menú configuración, desplegara el menú de opciones.
2. Seleccionar *administrar usuarios*, se desplegará una tabla con la lista de los usuarios si existen.
3. Dar clic en el botón *agregar usuario.*
4. Llenar el formulario con la información solicitada del usuario.
5. Dar clic en el botón guardar.

Figura 9. Agregar usuario

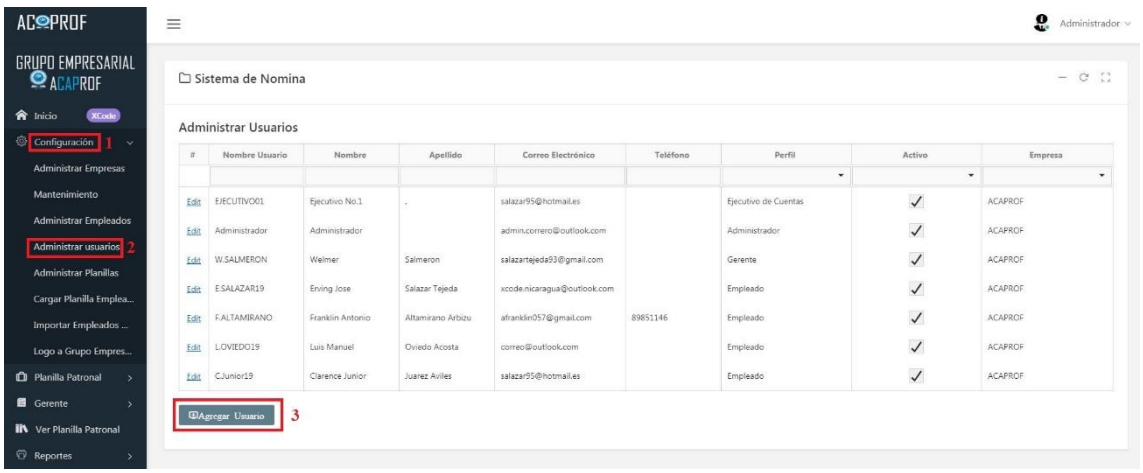
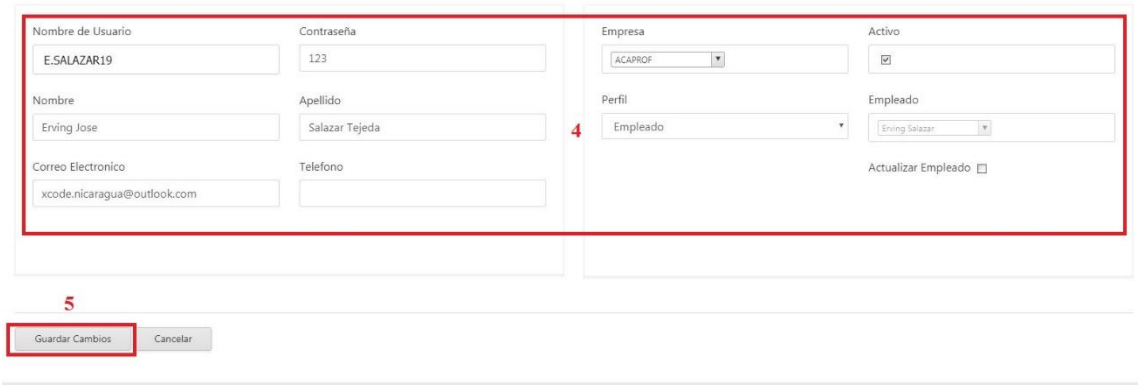


Figura 9.1. Formulario agregar usuario



18.7 Perfil ejecutivo de cuenta

Este usuario podrá será limitado el acceso por el administrador, cabe señalar que podrá agregar empleados y usuarios solo de las empresas que se le asignen.

El rol principal de este perfil de usuario es de administrar planillas, la información de los empleados y generar reportes con la información guardada en la base de datos.

18.7.1 Importar planillas de un Excel al sistema

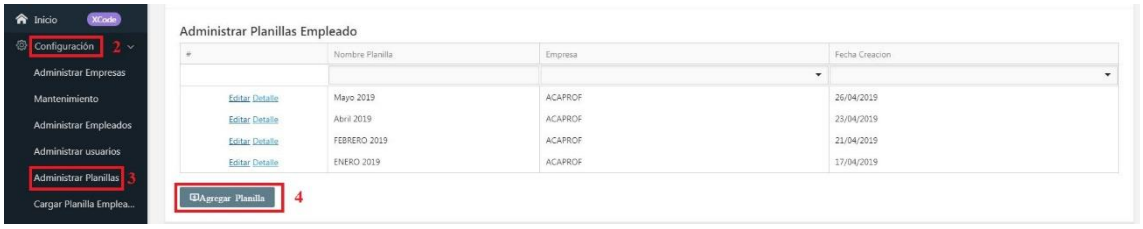
1. Preparar Excel para que se pueda subir al sistema
    a. Eliminar encabezados
    b. Eliminar tildes de los encabezados
    c. Asegurarse de que el código del empleado valla en la primera columna del Excel y que este sea el mismo que se guardó para el perfil de los empleaos dentro del sistema.
    d. Dejar una línea en blanco entre el último registro y el total.

Figura 10. Planilla en Excel

	EMPLEADO	NOMBRE	SUELDO_	DIAS_LAB_	I_SAL_	I_COMIS_	I_DEPRECIA	O_CEL_CONE	T_I_	INSS	IR	D_PLANAUTO	T_D_	SALNETO_
2	NCAC00001	ERVING JOSE SALAZAR TEJE	48,199.20	30	48,199.20	20,656.80	5,440.00	696.00	74,992.00	4,643.50	15,270.75	8,127.50	28,041.75	46,950.25
3	NCAC00002	FRANKLIN ANTONIO ALTAMIRA	38,899.00	30	38,899.00	16,671.00	5,440.00	696.00	61,706.00	3,813.13	11,534.06	5,786.78	21,133.97	40,572.03
4	NCAC00003	LUIS MANUEL OVIEDO ACOST	14,206.00	30	14,206.00	15,058.36	5,440.00	696.00	35,400.36	2,169.02	4,592.17	0.00	6,761.19	28,639.17
5														
6			101,304.20		101,304.20	52,386.16	16,320.00	2,088.00	172,098.36	10,625.65	31,396.98	13,914.28	55,936.91	116,161.45
7														
8	TOTAL EMPL	3												

2. En el sistema seleccionar en menú configuración, se desplegará el menú con varias opciones.
3. Dar clic en *administrar planillas,* abrirá una nueva página con las listas de las planillas creadas.
4. Dar clic en el botón *agregar planilla.*

Figura 10.1. Agregar planilla

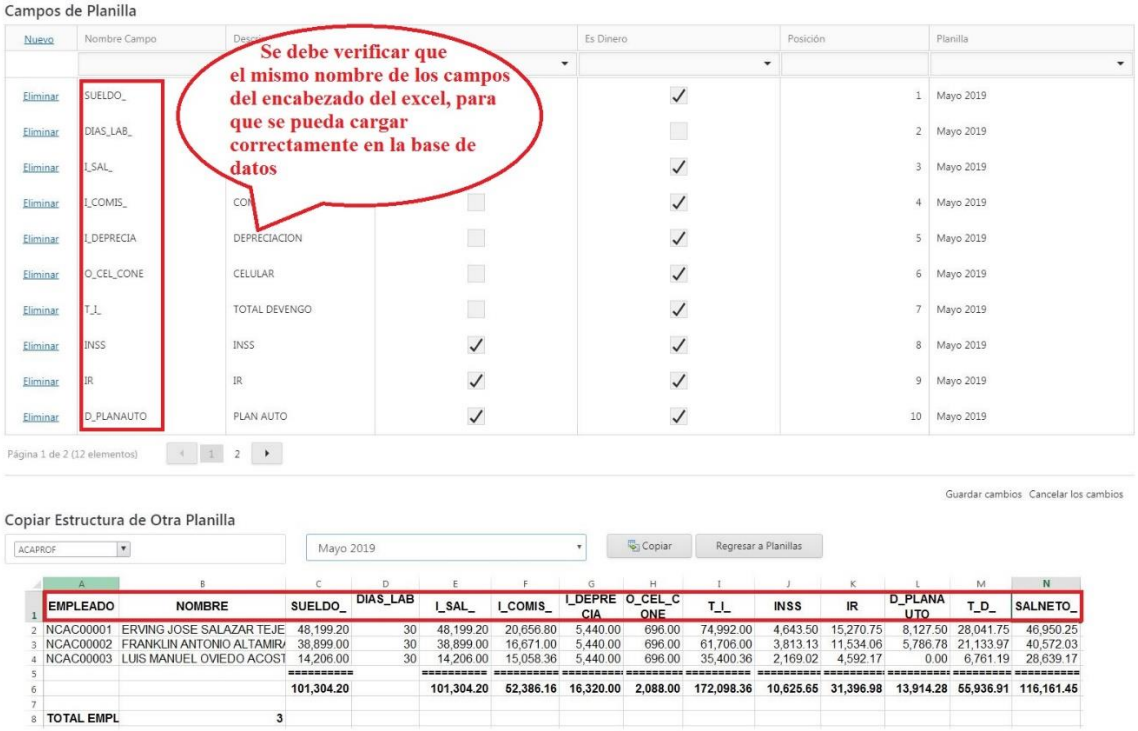


5. Llenar el formulario con el nombre de la planilla, esta corresponderá al mes y el tipo de planilla que corresponde.
6. Dar clic en el botón *Crear planilla.*, se abrirá una nueva página donde le daremos formato a la plantilla.

Figura 10.2. Crear Planilla

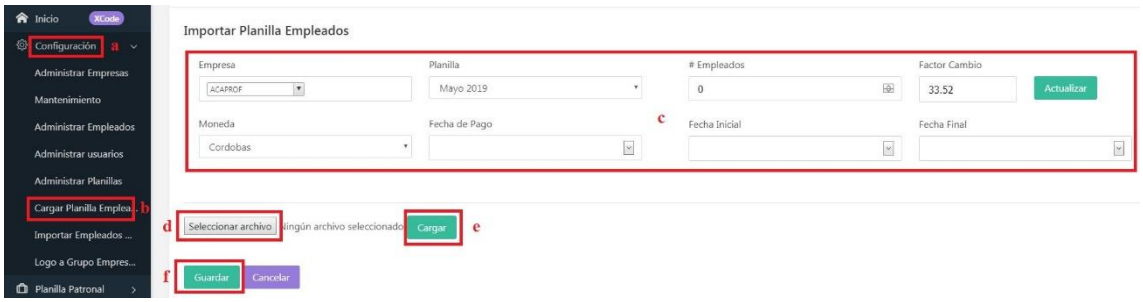
7. Copiar los campos del encabezado del Excel, o simplemente copiar un formato de una planilla del mes anterior y verificar que sea el mismo encabezado del Excel actual.

Figura 10.3. Crear planilla con el mismo formato del encabezado del Excel



8. Cargar planilla empleado.
    a. Seleccionar configuración en el menú.
    b. Dar clic en *cargar planilla*
    c. Llenar el formulario con las fechas de la planilla, el tipo de moneda y la planilla a la que se le cargara la información.
    d. Seleccionar el archivo de Excel.
    e. Dar clic en el botón *cargar*.
    f. Dar clic en el botón Guardar.

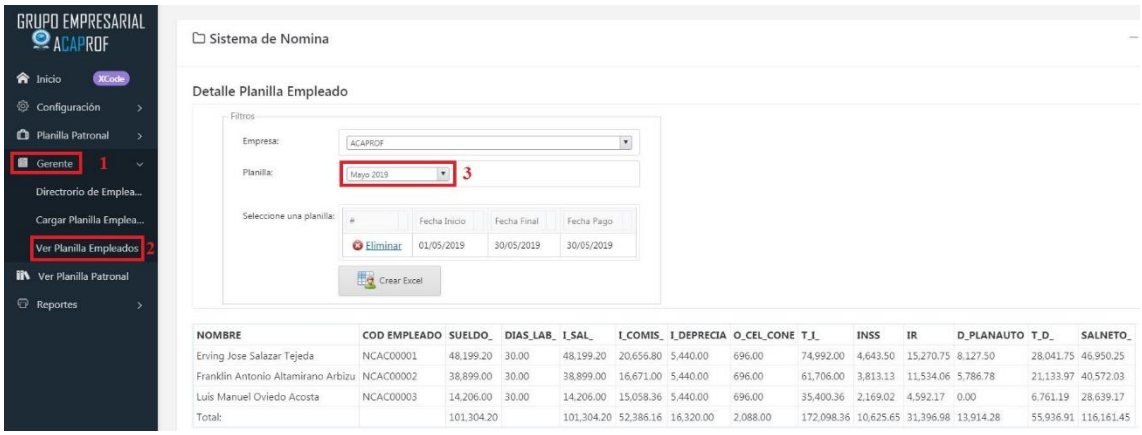
Figura 10.4. Cargar planilla



18.7.2 Visualizar planillas

1. Seleccionar en el menú *Gerente*.
2. Dar clic en *ver planilla empleado.*
3. Seleccionar la planilla del mes que se quiera visualizar.

Figura 11. Visualizar planillas

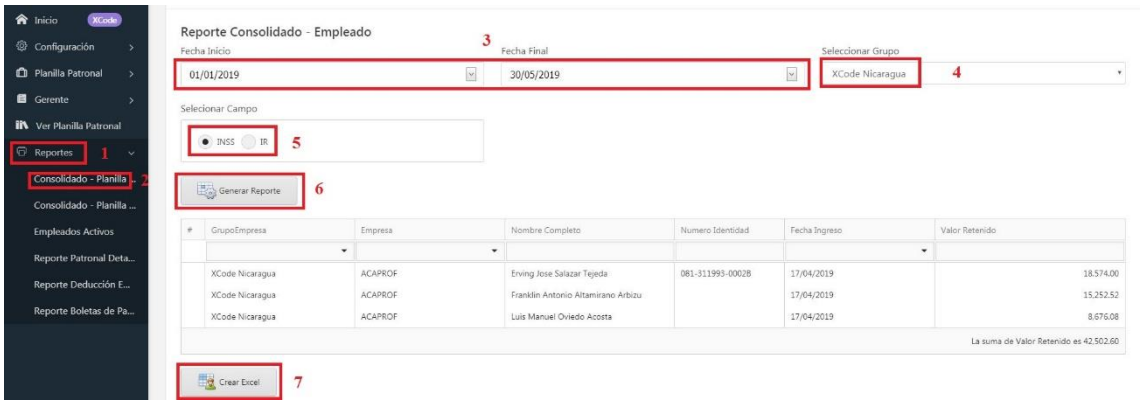


18.7.3 Reportes

18.7.3.1 Consolidado de deducciones

1. Seleccionar el menú *Reportes*, desplegara el menú de opciones.
2. Seleccionar el menú *Reporte Consolidado de Empleado* por grupo empresarial.
3. Seleccionar rango de fechas.
4. Seleccionar el grupo empresarial.
5. Elegir que deducción se quiere visualizar (INSS, IR).
6. Clic en el botón *Generar Reporte*
7. Si se quiere exportar en Excel, dar clic en el botón *Crear Excel*

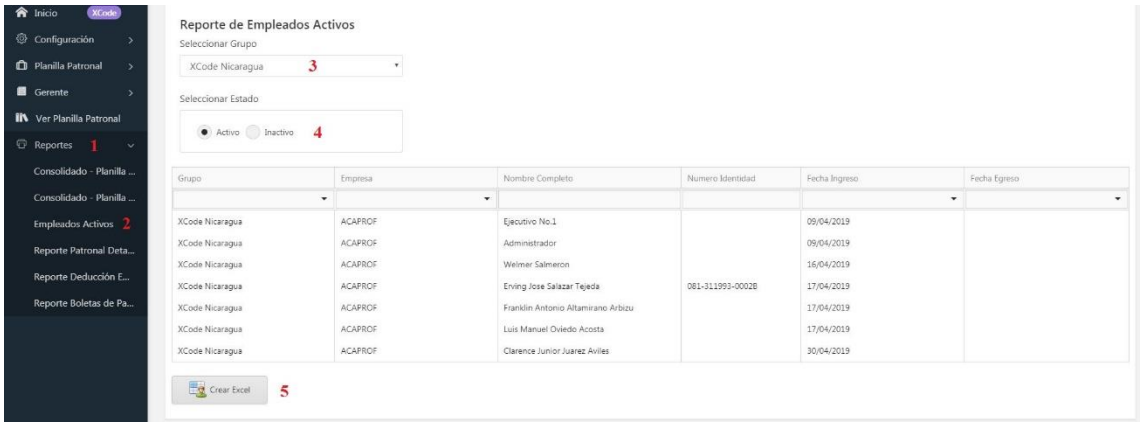
Figura 11. Reporte consolidado de deducciones de empleado por grupo empresarial



18.7.3.2 Reporte de empleados activos e inactivos por grupo empresarial

1. Seleccionar el menú *Reportes*, desplegara el menú de opciones.
2. Seleccionar el menú *empleados activos*.
3. Seleccionar el grupo empresarial.
4. Elegir el estado de los empleados.
5. Si se quiere exportar en Excel, dar clic en el botón *Crear Excel*

Figura 12. Reporte de empleados activos por grupo empresarial



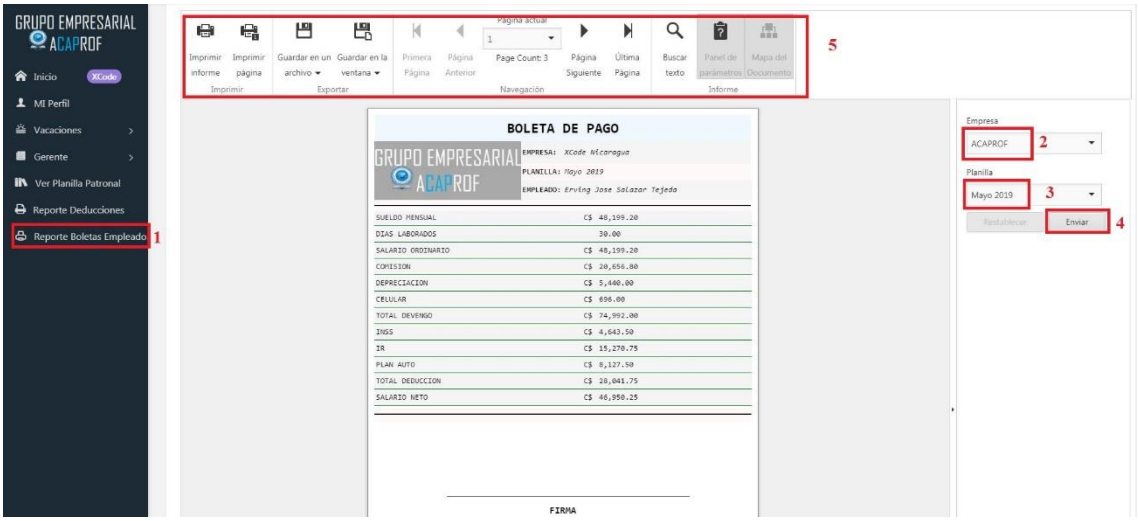
18.8 Perfil Gerente

18.8.1 Reporte de boletas de pagos de empleados masivos por empresa

1. Seleccionar el menú *Reportes boletas de pagos*.
2. Seleccionar la empresa.
3. Seleccionar la planilla.

4. Dar clic en el botón *enviar*, automáticamente se cargarán todas las boletas de pago de los empleados que pertenecen a la empresa según la planilla seleccionada.
5. En la parte superior aparecerá una cinta de opciones, donde podrá elegir entre descargar en diferentes formatos, imprimir, etc.

Figura 13. Reporte de boletas de pagos de todos los empleados por empresa.



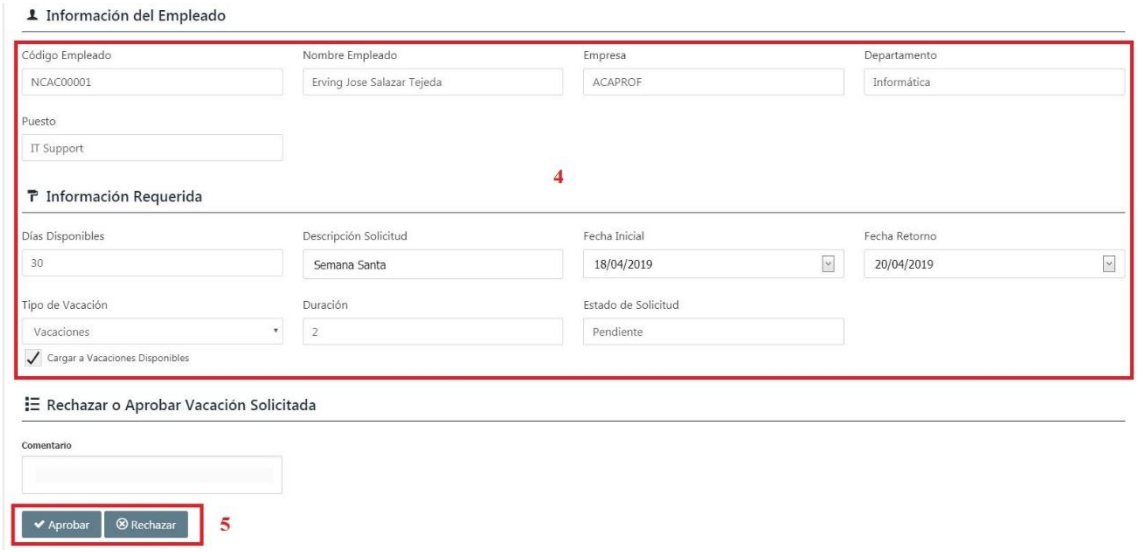
18.8.2 Aprobar o rechazar solicitud de vacaciones

1. Seleccionar el menú *vacaciones*, se desplegará una lista.
2. Seleccionar de la lista la opción *gestionar vacaciones*, se cargará una página con una tabla listando las vacaciones solicitadas por los empleados.
3. Seleccionar *ver detalle*, se abrirá una nueva pantalla con la información de la solicitud de vacación.

Figura 13. Gestión de solicitudes de vacaciones.

4. Verificar la información.
5. Clic en el botón Aprobar o Rechazar según lo decidido.

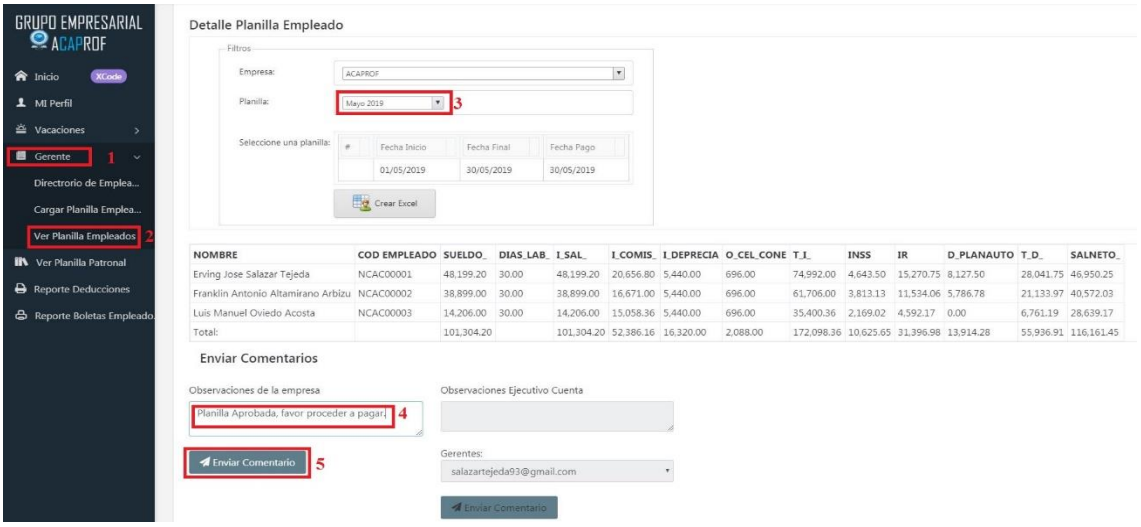
Figura 13.1 Aprobación o rechazo de solicitudes de vacaciones.



18.8.3 Aprobar planillas

1. Seleccionar en el menú *Gerente*, se desplegará una lista de opciones.
2. Seleccionar de la lista la opción *ver planilla empleado.*
3. Seleccionar *la planilla del mes*.
4. Llenar el campo de observaciones.
5. Dar clic en el botón *enviar comentario*.

Figura 14. Aprobar planillas.

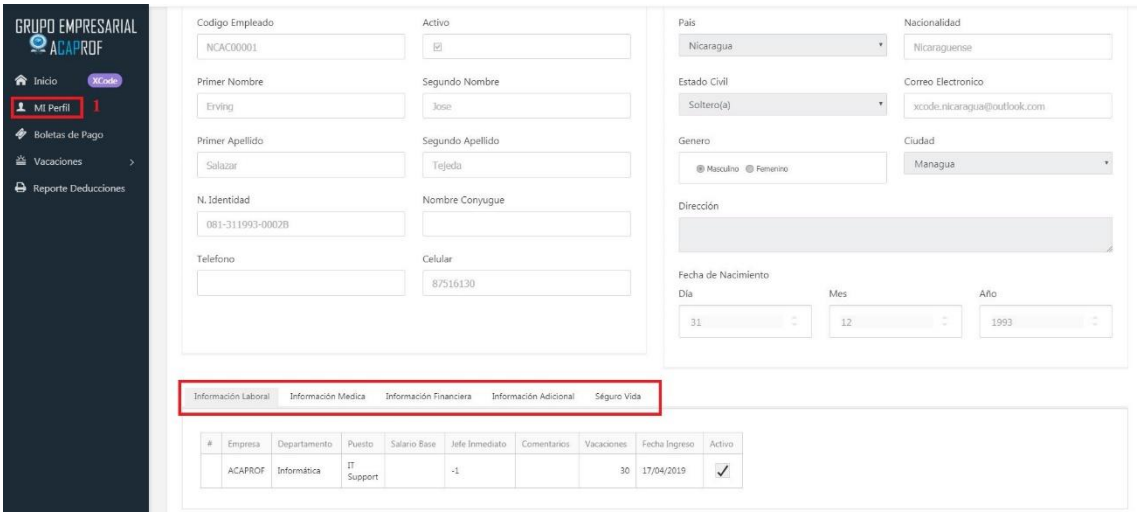


18.9 Perfil Empleado

18.9.1 Ver perfil

1. Seleccionar el menú *Perfil*, se cargará una pantalla con la información personal y laboral

Figura 15. Perfil de empleado.

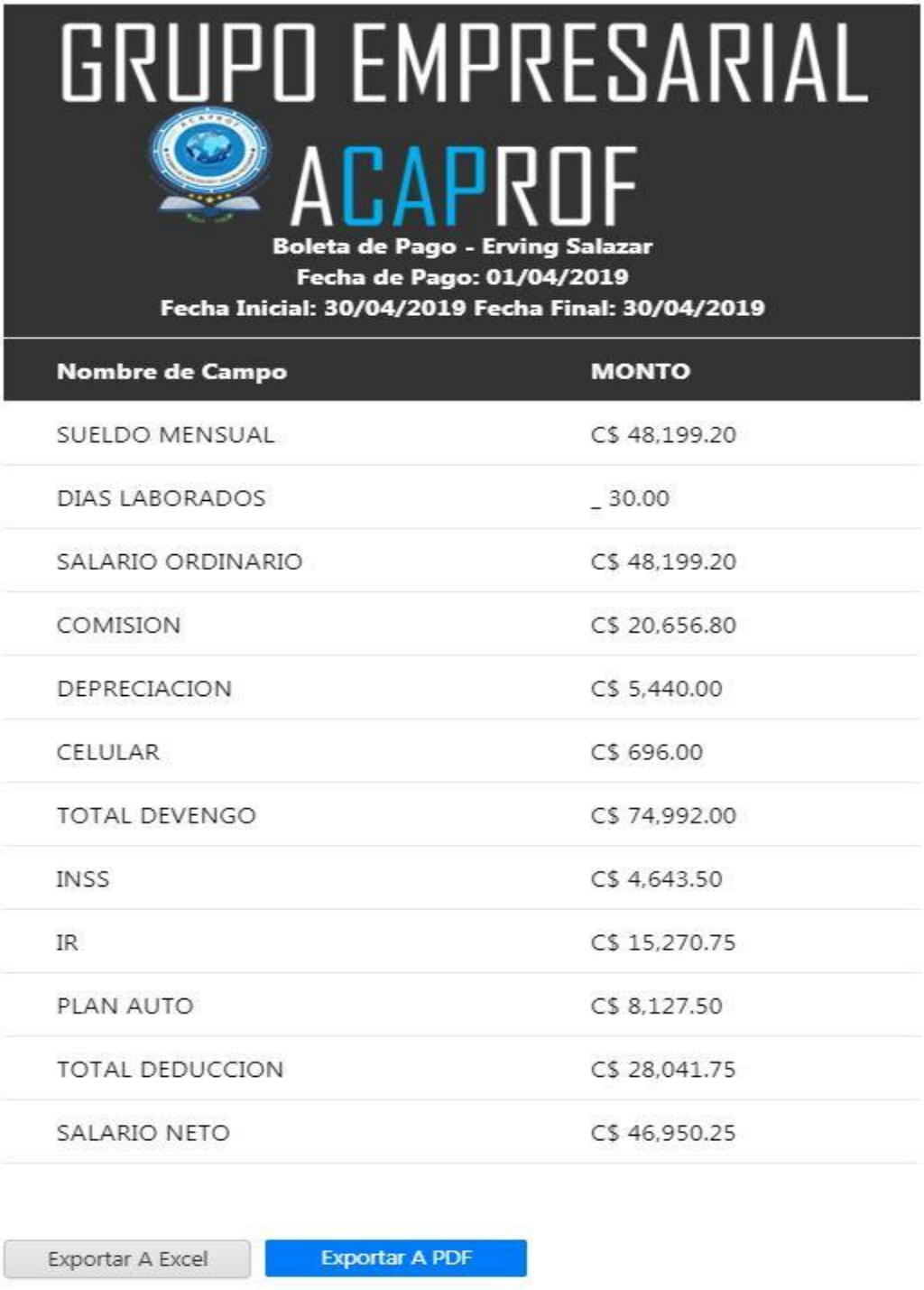


18.9.2 Ver Boletas de pago

1. Seleccionar el menú *Boleta de pago*, se cargará una pantalla con el historial de todas las boletas.
2. Según el mes que se desea, seleccionar la opción *ver*, se cargara una pantalla con la boleta de pago de la planilla.

Figura 16. Ver Boleta de pagos.

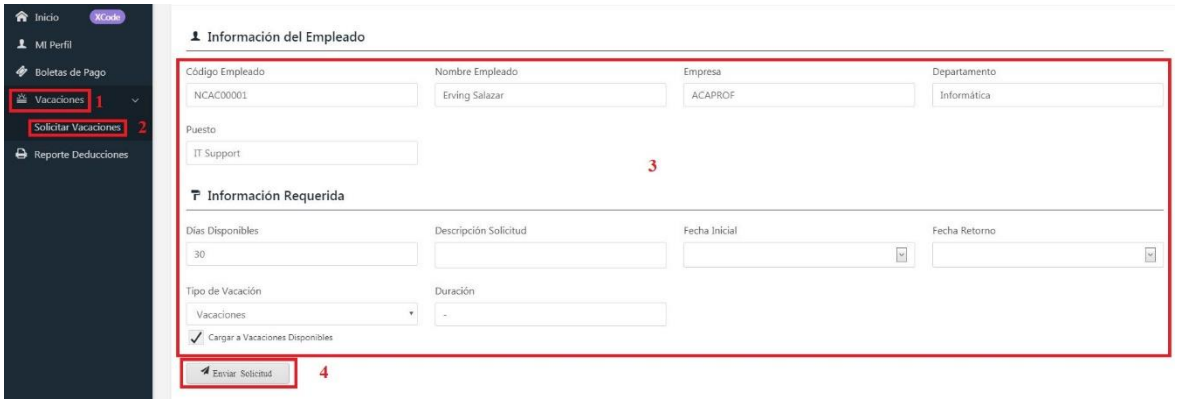
Figura 16.1. Boleta de pagos.



18.9.3 Solicitar vacaciones

1. Seleccionar el menú *Vacaciones*, desplegara una lista de opciones.
2. Seleccionar la opción *Solicitud de vacaciones*, cargara una nueva página con el formulario de solicitud.
3. Llenar el formulario con la información solicitada.
4. Dar clic en el botón *Enviar solicitud.*

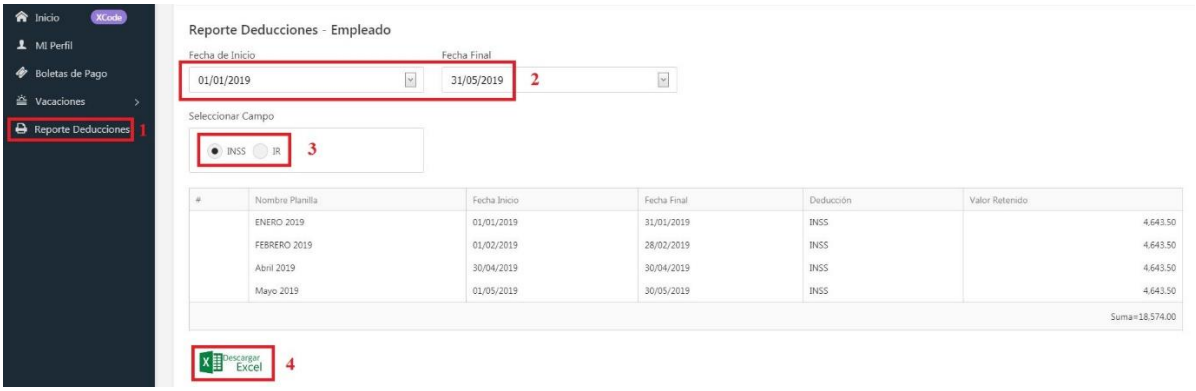
Figura 17. Solicitar Vacaciones.



18.9.4 Reporte de deducciones

1. Seleccionar el menú *Reporte deducciones*, cargara una pantalla con formulario.
2. Seleccionar rango de fechas.
3. Elegir deducción (INSS, IR).
4. Si se quiere descargar en Excel dar clic en el botón *Descargar Excel.*

Figura 18. Reporte deducciones.



18.10. Conclusión

Los roles asignados a los ejecutivos y gerentes tienen acceso de hacer otras gestiones las cuales no se detallaron para evitar redundancia ya que se detallaron en la parte de lo que hace el administrador, la diferencia de estos perfiles y acciones reside en que estos usuarios solo manejaran información de la empresa a la que pertenecen al contrario del administrador que maneja y puede modificar todas las funciones del sistema de todas las empresas a las que se le llevan las planillas.

Nota: *Cuando se realiza cualquier solicitud o aprobación, el sistema tiene configurado para que se envié un correo electrónico al usuario encargado de gestionar cada proceso, así mismo llega una copia al ejecutivo de cuenta que tiene asignado la empresa, esta con el objetivo de que les dé seguimiento a las solicitudes.*

19. RECURSOS

19.1. Recursos Humanos

Se constará con las personas involucradas en la realización y finalización de este proyecto, así como del encargado de facilitarnos toda la información disponible.

19.2. Recursos Físicos

Las herramientas Hardware necesarias para el diseño del sistema, serán las computadoras personales del recurso humano involucrado en el desarrollo del proyecto.

Las herramientas software necesarias para el diseño del sistema, será implementado bajo el sistema operativo Windows 7 Ultimate, Entorno de programación Microsoft Visual Studio 2014, Gestor de Bases de Datos SQL Server 2012 Express, Lenguaje de Programación Visual Basic.

19.3. Recursos Financieros

No se contará con ningún recurso económico para la finalización de este proyecto.