



Facultad Regional Mártires de Artemisa

**Trabajo de diploma para optar por el título de  
Ingeniero en Ciencias Informáticas**

Título: Sistema de Nóminas para las oficinas de  
economía y finanzas del MININT de la Provincia  
de Artemisa

**Autor(es):** Ariel Castro García

**Tutores:** Ing. Yoleidy Socorro Álvarez  
Ing. Leosmel Zayas Castillo

Artemisa, junio 2012

# Sistema de Nóminas

---

---

## DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter no exclusivo. Para que así conste firmo la presente a los \_\_\_\_ días del mes de Marzo del año 2012.

---

Ariel Castro García  
**Firma del autor**

---

Ing.Yoleidy Socorro Álvarez  
**Firma del tutor**

---

Ing.Leosmel Zayas Castillo  
**Firma del tutor**

*"Una de las cosas más fascinantes de los programadores es que no puedes saber si están trabajando o no sólo con mirarlos. A menudo están sentados aparentemente tomando café, chismorreando o mirando a las nubes. Sin embargo, es posible que estén poniendo en orden todas las ideas individuales y sin relación que pululan por su mente".*

*Charles M. Strauss*

## Agradecimientos

*Agradecer a todo aquel que creyó en mí cuando todo parecía imposible.*

*A mi familia por apoyarme siempre incondicionalmente.*

*A mis padres y a mi mujer por siempre estar ahí.*

*A mis tutores Leosmel y Yoleidy por su ayuda incondicional.*

*A la profe Yeilín por brindarme su ayuda en todo momento.*

*En general mi más sincero agradecimiento para todo aquel que me ayudo a llegar a este momento.*

## Dedicatoria

*Este trabajo va dedicado a mi mujer, por darme el más grande que poseo y a mi hijo Diego por ser ese grandísima tesoro.*

# Sistema de Nóminas

---

---

## Resumen

La gestión de la información se ha convertido en una necesidad en la actualidad debido a los grandes volúmenes de información que se genera en las empresas e instituciones modernas.

Los sistemas informáticos para la gestión de la información han demostrado su veracidad en todos los sectores de la sociedad. Consecuentemente se decidió desarrollar un sistema para la gestión de toda la información que genera un proceso de nóminas.

El presente trabajo de diploma abarca la investigación, desarrollo y resultados del Sistema de Nóminas para las oficinas de economía y finanzas del MININT (Ministerio del interior). El mismo surge a raíz de la necesidad de perfeccionar e informatizar los procesos que se llevan a cabo en la realización de la nómina en aras de mejorar la disponibilidad, calidad y seguridad de la información acerca de cada trabajador y su nómina.

El trabajo incluye el estudio del estado del arte, las herramientas y metodologías utilizadas, las características del sistema, planificación, implementación y la estrategia de pruebas que se llevó a cabo.

**Palabras claves:** Información, gestión de información, nómina.

# Sistema de Nóminas

---

---

## Índice

INTRODUCCION.....	1
CAPÍTULO 1. Fundamentación teórica .....	9
1.1 Conceptos básicos relacionados con el problema planteado .....	9
1.3 Metodología, técnicas y herramientas utilizadas para el desarrollo .....	12
1.3.1 Selección de la Metodología de Desarrollo a utilizar .....	14
1.3.2 Herramienta de modelado .....	15
1.3.3 Sistema operativo a utilizar .....	16
1.3.4 Lenguajes de programación .....	16
1.3.5 Entorno de Desarrollo Integrado.....	18
1.3.6 Selección del Entorno de Desarrollo Integrado. ....	19
1.3.7 Sistema Gestor de base de datos .....	21
CAPÍTULO 2: Características, análisis y diseño del sistema.....	23
2.1 Características y principales funcionalidades del sistema. ....	23
2.2 Planificación del proyecto por roles. ....	24
2.3 Modelo de dominio.....	26
2.4 Lista de reserva del producto (LRP).....	27
2.5 Historias de usuario y tareas de ingeniería. ....	31
2.6 Plan de release. ....	45

# Sistema de Nóminas

---

---

2.7 Arquitectura de software. ....	45
2.8 Diseño con metáforas. ....	47
2.9 Diagrama de componentes. ....	48
CAPÍTULO 3. Validación implementación. ....	49
3.1 Casos de pruebas. ....	50
3.2 Casos de pruebas unitarias. ....	63
3.3 Resultados y funcionalidades obtenidas. ....	64
3.4 Aporte social y económico. ....	64
REFERENCIAS BIBLIOGRÁFICAS. ....	68
Anexos. ....	69



## INTRODUCCION

La posesión y el control de la información ha sido uno de los medios más poderosos de los gobiernos para hacer llevar sus políticas. Así, los estados totalitarios y autoritarios buscan ser poseedores únicos de la información para promover el acatamiento de las políticas. La información tiene por objetivo dar a conocer los hechos de manera efectiva e imparcial.

Para las ciencias informáticas, la información es un conocimiento explícito extraído por personas o sistemas expertos como resultado de la interacción con el entorno o percepciones sensibles del mismo entorno. En principio la información, a diferencia de los datos o las percepciones sensibles, tienen estructura útil que modificará las sucesivas interacciones del ente que posee dicha información con su entorno.

En la informática uno de los métodos de almacenamiento de la información más usados son las base de datos, estas entran en utilidad generalmente cuando se habla de una cantidad de información bastante amplia. Las mismas han pasado a ser una de las herramientas más ampliamente difundidas en la actual sociedad de las tecnologías de la información, utilizadas principalmente en la recuperación y almacenamiento de información en todos los campos de la sociedad actual. El uso de las bases de datos o de sistemas parecido a estas, data de unos cuantos años atrás, pero no fue hasta la década de los noventa que su uso fue prácticamente imprescindible. En los principios de esta década surge el lenguaje SQL, que es un lenguaje programado para consultas.

Si bien es verdad que los sistemas de bases de datos, son una de las herramientas más importantes para el desarrollo de cualquier área del saber, el contexto histórico ha demostrado que son de una gran influencia en el progreso de los países a nivel mundial, debido a la gran ayuda que brindan en la toma de decisiones permitiendo un almacenamiento y acceso confiable, eficiente y práctico en el uso de la

## Sistema de Nóminas

---

---

información que se produce. En la actualidad las bases de datos juegan un papel muy importante en las tecnologías de la información, ya que son la principal herramienta en los importantes sistemas de información.

Las Tecnologías de Información y los Sistemas de Información han cambiado la forma en que trabajan las organizaciones actuales. Las tecnologías de la Información han sido catalogadas como la unión y convergencia de las telecomunicaciones, la computación, y la técnica para el procesamiento de datos, siendo sus principales componentes: el factor humano, los contenidos de la información, el equipamiento, la infraestructura, el software, los mecanismos de intercambio de información, los elementos de política y regulaciones, además de los recursos financieros.

En la actualidad se dice que se encuentran frente a una sociedad de información global emergente, que contiene una economía global que cada vez más tiene una dependencia de la administración, creación, y la distribución de la información a través de redes globales como Internet o de los sistemas informáticos. Muchas empresas están en proceso de globalización; es decir, se están convirtiendo en empresas globales interconectadas en red. El manejo y la realización de estos cambios estratégicos serían imposibles sin Internet, Intranets y otras redes globales de computación y de telecomunicaciones que constituyen un sistema nervioso central de las empresas globales de hoy.

Se dice que el uso de las nuevas tecnologías de la información es uno de los principales motores de la sociedad actual. Las infocomunicaciones y su fuerte impacto en los países del primer mundo, ha hecho que durante los últimos años comenzaran a plantearse la existencia de la sociedad de la información. Ejemplo de esto fue en 1993 Japón crea el Consejo de Telecomones, mientras en Europa y Canadá en 1995 se planteaban proyectos sobre autopistas de la información. No solo los países desarrollados tomaron la iniciativa en 1995-96 en África se lanza la iniciativa de la sociedad de la información africana (AISI), es decir, en uno de los

## Sistema de Nóminas

---

---

continentes más pobres se manifiestan las mismas necesidades que en los países más desarrollados.

Basándose en lo antes mencionado sobre los países desarrollados estos procedieron a elaborar sus programas, el Grupo de los 8 anunció su Programa "Iniciativa para la Sociedad Global de la Información". A partir de ahí muchos países comenzaron a lanzar proyectos sobre la cultura de la información para todas las esferas del país, educación, salud, economía entre otras. Cuba no fue una excepción un poco después comienzan proyectos para informatizar la sociedad cubana. El Programa para el Fomento de la Cultura Informática (IS-E) en Cuba, parte de que el acceso a la educación desde el preescolar a la universidad es un derecho de todo ciudadano, garantizado por el estado y que en estos momentos el nivel mínimo de escolarización es superior al 9no. grado, así las cosas, la Informatización de la Sociedad implica la cultura Informática de todos sus miembros y esa cultura sólo tiene sentido en la Infosociedad. En la Sociedad Cubana el Sistema de la Cultura Informática tiene como objetivo principal al personal, sea este un estudiante, un trabajador no importa si es industrial o agrícola , un funcionario, un ejecutivo, un especialista, un profesional, un profesor e inclusive una ama de casa o un jubilado es decir toda la sociedad pertenece y por ello forman parte de los mismos todas las instituciones pertenecientes al proceso regular de educación (preescolar, primaria, secundaria, preuniversitario, universitario, postgrado, de capacitación ramal y territorial (centros de adiestramiento y superación de los Ministerios, Gobiernos Territoriales, Casas de Cultura y Joven Club de Computación) y los órganos de difusión masiva.

Bajo a estas ideas y junto a las transformaciones sociales que se llevaban a cabo por el programa de la batalla de idea, surge la universidad de las ciencias informáticas (UCI).La UCI es una universidad distinta a las otras existentes en el país a parte de su modelo académico posee un sistema de producción, donde se

## Sistema de Nóminas

---

---

realizan proyectos que algunos ayudan con la informatización del país y algunos de colaboración internacional.

La Idea del Comandante en Jefe sobre la UCI se refleja en esta frase “La idea es convertir la informática en una de las ramas más productivas y aportadoras de recursos para la nación. Es el empleo a fondo de la inteligencia y del capital humano que tienen y principalmente del que se puede crear casi como espina dorsal de la economía” .La Infraestructura Productiva de la UCI coordina la actividad productiva y brinda servicios para asegurar el correcto desarrollo y terminación de proyectos, productos o servicios.

La producción se concentra en el desarrollo de proyectos y entre los resultados más importantes se destacan algunos en las esferas de salud, educación, software libre, automatización, bioinformática, teleformación, sistemas legales, realidad virtual, procesamiento de imágenes y señales. En el 2005 fue creada la empresa comercializadora de todos los productos y servicios informáticos que se desarrollen en la Infraestructura Productiva de la UCI.

Entre los estudiantes de la UCI se encuentran grupos de estudiantes que son preparados para la producción de software y otras soluciones informáticas en diversas organizaciones, como pueden ser las Fuerzas Armadas Revolucionarias o el MININT. Los estudiantes pertenecientes a estas organizaciones desarrollan mientras estudian, software que ayudan a automatizar y desarrollar a los mismos, ejemplos de esto pasa en el MININT. Con la nueva distribución político administrativa creada en cuba , se divide la provincia de la Habana en Artemisa y Mayabeque, al entrar en vigor esta distribución también se dividen todos los órganos de dirección existentes en el país .Excepción de esto no fue el MININT (MININT) que se dividió en dos direcciones. La dirección de informática comunicación y cifras (DICC) es la encargada de la realización de software en el MININT. Al realizarse la distribución muchos software quedaron inutilizados, ya que

## Sistema de Nóminas

---

---

muchas bases de datos e información de estos programas se encontraban desactualizados o simplemente, no se correspondían con la necesidad que tenía la provincia.

El Sistema de Nóminas que actualmente se utiliza no se corresponde con las necesidades que requieren los usuarios que lo trabajan pues necesitan de más información que la que el sistema ofrece .La creación de estos sistemas que actualmente se encuentran en implementación y desarrollo en el MININT, forman parte de un sistema integrado el cual va destinado principalmente a la protección de la información ,que es uno de los principales problemas que pueden ocurrir , la pérdida de la misma puede ocurrir de forma accidental o adrede , por catástrofes naturales o por robo de información de algún personal no autorizado , también el sistema va destinado a facilitar el trabajo de los usuarios , ya sea informático o no , militares o civiles que trabajen para el ministerio.

Analizado lo anterior se determina el siguiente **problema de Investigación**

¿Cómo perfeccionar la confiabilidad y seguridad de la información en el proceso de Nóminas realizado por las oficinas de economía y finanzas del MININT de la provincia de Artemisa?

Se identifica como **objeto de estudio**: Sistemas de gestión de la información. Dentro del objeto de estudio anteriormente mencionado se enmarca el **campo de acción**: Sistemas de Gestión de Información para el proceso de nóminas en el MININT de la provincia de Artemisa.

Se define como **objetivo general** Desarrollar una aplicación que garantice la correcta gestión de la información así como las operaciones necesarias en el proceso de nóminas en el MININT de la provincia de Artemisa.

**Idea a defender**: Con el desarrollo del Sistema de Nóminas se garantizará una correcta gestión de la información en las oficinas de economía y finanzas del MININT.

# Sistema de Nóminas

---

---

Definido el objetivo general, se desglosan los siguientes **Objetivos Específicos:**

1. Estudiar las herramientas, metodologías y tecnologías definidas para desarrollar la una aplicación y su base de datos para el sistema de nómina.
2. Diseñar una aplicación y su base de datos para el Sistema de Nóminas
3. Implementar una aplicación y su Base de Datos para el Sistema de Nóminas.
4. Validar mediante pruebas funcionales los resultados obtenidos con la solución.

Para darle cumplimiento a los objetivos específicos anteriormente mencionados, se han trazado las siguientes **Tareas de la Investigación:**

- Estudio del marco teórico que fundamenta el objeto de investigación
- Selección de las principales herramientas a utilizar.
- Diseño del Modelo Lógico y Físico de la base de datos.
- Estudio de la metodología de desarrollo de software y herramientas de desarrollo a usar en el proceso definido por el proyecto.
- Identificación de los requerimientos de la solución a desarrollar.
- Diseño de la solución de acuerdo a los modelos realizados y los requerimientos identificados.
- Implementación de la aplicación.
- Ejecución de los casos de prueba de aceptación para asegurar la calidad del resultado.

## **Métodos**

### **Métodos Teóricos**

#### **Histórico – Lógico**

Se realizó una amplia revisión bibliográfica en diferentes artículos digitales, sitios web y documentos actuales, con el objetivo de conocer acerca de los sistemas información y sistemas informáticos. Surgiendo así la necesidad del desarrollo de un sistema de información para el área de finanzas en el MININT en Artemisa y los diferentes componentes para la implementación del mismo.

**Inductivo – deductivo:** En función del problema a resolver planteado, se pone en práctica un grupo de conocimientos sobre las tendencias actuales y características generales de los procesos relacionados.

**Modelado:** Con el objetivo de realizar el análisis y diseño del software y posteriormente la implementación

### **Métodos Empíricos**

**Entrevista:** Se realizó a miembros de la dirección del MININT específicamente a los trabajadores del departamento de economía y finanzas, con el objetivo de recopilar información sobre el proceso de gestión de nómina.

De un total de 34 personas que trabajan en los departamentos mencionados, se le realizó la entrevista a 18 personas, las cuales son las involucradas en la primera versión de la solución propuesta. El 100% de los entrevistados aportó como resultado la identificación de los aspectos del proceso de nómina que debían perfeccionar.

# Sistema de Nóminas

---

---

Las variables identificadas en esta investigación fueron las siguientes: como **variable independiente:** Sistema de nóminas del MININT y las **variables dependientes:** seguridad y confiabilidad.

## Posibles resultados:

- Informe detallado con toda la base teórica sobre la cual se sustenta la solución propuesta.
- Aplicación Sistema de Nóminas que automatice el proceso de nóminas en el MININT

## Estructura del Documento

Capítulo 1: Fundamentación Teórica. Se realizarán explicaciones de los distintos conceptos utilizados en el documento. Este capítulo incluye un estado del arte del tema tratado a nivel internacional, nacional y de la universidad. Se analizarán también las diferentes tecnologías, lenguajes y metodologías de desarrollo de software empleadas en la realización del sistema.

Capítulo 2: Características, análisis y diseño del sistema. Se realiza una propuesta del sistema, se describe cómo debe funcionar y se destaca sus características distintivas; además, se especifican sus Requisitos Funcionales y No Funcionales. Se realiza el análisis y diseño del sistema donde se especifican los principales artefactos generados en las primeras fases.

Capítulo 3: Adquisición y validación de los resultados del sistema. Incluye toda la información relacionada con el desarrollo del sistema. Además, se muestran la información adquirida a raíz de la realización de las pruebas al sistema a través de los casos de prueba de aceptación.



## **CAPÍTULO 1. Fundamentación teórica**

En el presente capítulo se realiza una descripción con relación a conceptos básicos relacionados con el dominio del problema a resolver. Se aborda el tema de las nóminas y cómo se realiza, además de los sistemas de gestión de la información. Se analizan igualmente las principales herramientas y tecnologías a utilizar para el desarrollo de los procesos relacionados con el diseño e implementación del Sistema de Nóminas para MININT en Artemisa.

### **1.1 Conceptos básicos relacionados con el problema planteado**

#### **¿Qué es la nómina?**

La **nómina es el documento** que se entrega mensualmente a todos los trabajadores en el que aparece el detalle del salario que recibe, junto con las deducciones que se le practican de dicho salario, bien sea por descuentos obligatorios marcados por la legislación vigente, bien sea por otro tipo de descuentos como anticipos, o créditos por pagar.

#### **Proceso de Nómina**

- Procesamiento y pago de las nóminas
- Cumplimiento de los deberes formales relacionados con las nóminas
- Proceso de enteramiento de las retenciones y los aportes patronales
- Cálculo de las prestaciones sociales

# Sistema de Nóminas

---

---

- Procesamiento de los datos para los registros contables relacionados con la nómina y acumulaciones de pasivos del personal
- Procesamiento de los aportes de empleados y patrono a la Caja de Ahorros
- Cálculo de las utilidades del personal

## **Pre nóminas**

Permite consolidar la información en el departamento económico, manteniendo la identidad de cada área de trabajo para facilitar la comprensión de los datos.

## **Retenciones**

Contiene parte del salario devengado del trabajador, que no se le entrega con el pago y que se retiene con el propósito de liquidar obligaciones.

## **1.2 Estado del arte**

### **Sistemas de nóminas en el mundo**

En el mundo existen sistemas que se han creado con el objetivo de apoyar y automatizar los procesos de nóminas que se llevan a cabo, haciendo énfasis en el análisis y el almacenamiento de los datos que se obtienen en las mismas.

Entre ellos se encuentra:

**TRESS:** Es la única solución en el mundo que desde su diseño combina de manera transparente para el usuario final funcionalidad para la administración de recursos humanos, nómina y control de asistencia. Esta ventaja de diseño, aunada a la experiencia y enfoque de Grupo Treses Internacional en los mercados de México y América Latina, ha permitido que cientos de clientes obtengan substanciales ventajas competitivas.

**Asaca:** La característica fundamental de este Sistema de Nómina de Pagos de Personal, y Administración de Recursos Humanos, es la gran capacidad de Transacciones, totalmente programables y parametrizables por el Usuario.

## Sistema de Nóminas

---

---

El Usuario dispone de casi 1.000 Transacciones de Nómina que están divididas en nueve grupos, según su naturaleza. Por cada una de ellas, el usuario puede obtener una Consulta o un Reporte, por Fecha o por Trabajador.

**Aspel-NOI:** La principal característica de este sistema es que automatiza el control de los aspectos más importantes de la nómina, su fácil manejo y versatilidad ofrecen un cálculo exacto de las percepciones y deducciones de los trabajadores atendiendo los requerimientos específicos de la empresa. Calcula la retención de ISR, subsidio para el empleo y previsión social, entre otros, de acuerdo con la legislación fiscal y laboral vigente.

### **Sistemas de Nóminas en Cuba y la UCI**

A nivel nacional existen varios sistemas para la gestión de nóminas, cada sucursal empresaria cuenta con su propio sistemas de nóminas, ejemplo de esto es **SNBPA** usado en el banco popular de ahorro, **SISCOM** se le nombra al utilizado en Etecsa, los software anteriormente mencionado son pequeños software orientados específicamente a la empresa que los creo .

**Assets Premium** software de importación comprado recientemente por la UCI a una compañía italiana, el software tiene una notable mejora en comparación con el existente anteriormente, este software es capaz de integrar todos los modelos de nóminas y realizar los cálculos necesarios de una nómina.

Ninguno de los software mencionado anteriormente cumplen con el problema que presenta el MININT para la realización del proceso de nómina. El proceso de nóminas realizado en el MININT de la provincia de Artemisa cuenta con características especiales por lo cual se necesita la realización de un sistema único para ese centro.

## 1.3 Metodología, técnicas y herramientas utilizadas para el desarrollo

### Metodologías de Desarrollo

Los desarrolladores a la hora de realizar una aplicación necesitan saber cómo organizar las actividades para cada desarrollador por separado y para el equipo, definir qué artefactos deben ser creados y contar con una serie de criterios que permitan controlar y medir los productos que se obtienen, por lo tanto, se necesita de una metodología capaz de dirigir estas actividades y así convertir los requisitos de los usuarios en un producto *software*.

A continuación se brindan características de algunas metodologías de desarrollo que existen a fin de definir cuál será la que se usará para el desarrollo del sistema.

**Proceso Unificado de Desarrollo (RUP):** es una metodología de desarrollo de software que está basado en componentes e interfaces bien definidas, y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Es un proceso que puede especializarse para una gran variedad de sistemas de software, en diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto.

RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

Es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML, y trabajo de muchas metodologías utilizadas por los clientes. La versión que se ha estandarizado vio la luz en 1998 y se conoció en sus inicios como Proceso Unificado de Rational 5.0; de ahí las siglas con las que se identifica a este proceso de desarrollo.

**Scrum** es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

**Programación Extrema (XP):** Actualmente la mayoría de los programadores no piensan en una metodología de desarrollo a la hora de crear algún software, o sea tienen cierta tendencia de centrarse en cuestiones técnicas, hablar de lenguajes de programación, de técnicas de programación, de entornos de desarrollo o de editores de recursos. Pero se pasa por alto temas muy importantes como es la ingeniería de software, la manera en que se debe de hacer el software. Alrededor de cómo hacer software hay un gran número de teorías, propuestas.

El primer paso es conocer las metodologías más relevantes o buscar a alguien que las conozca, y en una situación ideal haber trabajado con varias de ellas. No hay metodología que funcione de manera universal, de hecho cada vez más las metodologías se conciben como 'marcos' metodológicos que es necesario ajustar para cada organización y tipo de proyecto. Realizar este ajuste es algo que necesita de una experiencia y un conocimiento previo. El problema con la implantación de una metodología es que no se suele tener una segunda oportunidad.

A la hora de seleccionar una metodología la primera decisión que se plantea es:

¿Una metodología ágil o una metodología guiada por plan? Puesto que la gran mayoría de proyectos se pueden beneficiar mucho del uso de una metodología ágil, pero indudablemente existen proyectos y entornos en los que es condición, generalmente impuesta por el cliente o la dirección de la empresa, que el proyecto se desarrolle con 'más control'.

Para plantearte el uso de una metodología ágil se debe ser capaz de asumir completamente el Manifiesto Ágil y ser capaces de hacer que sea el paradigma que guíe la gestión del proyecto, y desde luego es sumamente importante que logremos un sponsor. Tener un sponsor es vital en todo proyecto de implantación de una metodología, pero sobretodo es vital para implantar una metodología ágil, pues exige que se produzcan profundos cambios en la cultura tradicional relativa a la gestión de proyectos.

Poniendo de menos a más ágil, entre las metodologías más populares, se tiene las siguientes:

- eXtreme Programming
- Rational Unified Process
- Scrum

**SXP** es una metodología compuesta por las metodologías SCRUM y XP, que ofrece una estrategia tecnológica a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva donde se fomenta el desarrollo de la creatividad, se aumenta el nivel de preocupación y responsabilidad de los miembros del equipo y ayuda al líder del proyecto a tener un mejor control del mismo.

### 1.3.1 Selección de la Metodología de Desarrollo a utilizar

Realizado el análisis de las diferentes metodologías se llega a la conclusión que

dado el tiempo de realización del proyecto y la cantidad de miembros del equipo de desarrollo es mejor usar XP en conjunto con *Scrum*, pues RUP es una metodología enfocada al desarrollo de proyectos grandes, mientras que *Scrum* y XP se complementan el uno al otro de manera muy positiva. Las dos metodologías dan un círculo de retroalimentación de unos pocos segundos, donde los defectos son detectados y corregidos en segundos.

## **Lenguaje de Modelado.**

UML, por sus siglas en inglés, Unified Modeling Language: es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

### **1.3.2 Herramienta de modelado**

El modelado de sistemas de *software* es una técnica para tratar con la complejidad inherente a estos sistemas. El uso de modelos ayuda al ingeniero de *software* a "visualizar" el sistema a construir. Además, los modelos de un nivel de abstracción mayor pueden utilizarse para la comunicación con el cliente. Por último, las herramientas de modelado pueden ayudar a verificar la corrección del modelo.

Algunas de las herramientas CASE conocidas son el ArgoUML, Rational Rose, Visual Paradigm, Easy CASE, Xcase, CASE Studio 2, CASEWise entre otras. Dentro de las más utilizadas se encuentran el *Rational Rose* y el *Visual Paradigm*.

## **Selección de la Herramienta de modelado**

La herramienta utilizada es *Visual Paradigm*. Es una herramienta fácil de usar que soporta ingeniería inversa. Consta de una buena cantidad de productos o módulos para agilizar, optimizar y facilitar el trabajo durante la confección de un software, facilita la generación del modelo de la base de datos, no se inclina por ninguna metodología específica. Además, presenta:

Soporte para los diagramas de la versión de UML: UML 2.1.

Diseño centrado en casos de uso y enfocado al negocio, lo cual permite diseñar un *software* de mayor calidad.

Generación de código fuente en varios lenguajes de programación.

Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.

### **1.3.3 Sistema operativo a utilizar**

De las versiones de *Microsoft Windows* existentes se decide a usar:

*Microsoft Windows XP Service Pack 3 u otra versión superior*. Se utilizará este sistema para el cliente por ser un entorno fácil de trabajar, el más usados en los centros del MININT, con una amplia gama de *software*, documentación disponible en varios idiomas (entre ellas español).

### **1.3.4 Lenguajes de programación**

Es un idioma artificial diseñado para expresar computaciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear



programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

Está formado de un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación.

## **Selección del lenguaje de Programación**

El lenguaje de programación a utilizar será C# (*pronunciado si sharp en inglés*) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma.NET, que después fue aprobado como un estándar por la ECMA ISO.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi).

C#, como parte de la plataforma.NET, está normalizado por ECMA desde diciembre de 2001 (C# Language Specification "Especificación del lenguaje C#"). El 7 de noviembre de 2005 salió la versión 2.0 del lenguaje que incluía mejoras tales como tipos genéricos, métodos anónimos, iteradores, tipos parciales y tipos anulables. El 19 de noviembre de 2007 salió la versión 3.0 de C# destacando entre las mejoras los tipos implícitos, tipos anónimos y LINQ (*Language Integrated Query* -consulta integrada en el lenguaje).

## **Ventajas:**

- Lenguaje de programación orientado a objetos **simple, moderno y de propósito general.**

- Inclusión de principios de ingeniería de software tales como revisión estricta de los tipos de datos, revisión de límites de vectores, detección de intentos de usar variables no inicializadas, y recolección de basura automática.
- Capacidad para desarrollar componentes de software que se puedan usar en ambientes distribuidos.
- Portabilidad del código fuente
- Fácil migración del programador al nuevo lenguaje, especialmente para programadores familiarizados con C y C++.
- Soporte para internacionalización
- Adecuación para escribir aplicaciones de cualquier tamaño: desde las más grandes y sofisticadas como sistemas operativos hasta las más pequeñas funciones.
- Aplicaciones económicas en cuanto a memoria y procesado.

### **1.3.5 Entorno de Desarrollo Integrado.**

Un entorno de desarrollo integrado es una herramienta de soporte al proceso de desarrollo de *software* que integra las funciones básicas de edición de código, compilación y ejecución de programas con otras tales como:

- Editor con corrección sintáctica y coloreo de la misma al momento de codificar. Herramientas gráficas.
- Soporte integrado para la compilación y ejecución de programas.
- Relación entre errores de compilación y el código fuente, para facilitar su corrección.
- Opciones de *Debugging*.

## 1.3.6 Selección del Entorno de Desarrollo Integrado.

Se utilizará Visual Studio 2010 para lograr el desarrollo con el lenguaje Csharp. Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

### Framework

Un *framework* es una estructura *software* compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que se puede añadir las últimas piezas para construir una aplicación concreta.

### Las principales ventajas de la utilización de un *framework* son:

- ✦ El desarrollo rápido de aplicaciones. Los componentes incluidos en un *framework* constituyen una capa que libera al programador de la escritura de código de bajo nivel.
- ✦ La reutilización de componentes *software* al por mayor. Los *frameworks* son los paradigmas de la reutilización.
- ✦ El uso y la programación de componentes que siguen una política de diseño uniforme. Un *frameworks* orientado a objetos logra que los componentes sean clases que pertenezcan a una gran jerarquía de clases, lo que resulta en bibliotecas más fáciles de aprender a usar.

### Selección de los *Framework*

El .Net *Framework* en su versión 4.0.3 (traducido como “Marco de Trabajo”) es el componente fundamental de la plataforma *Microsoft .NET*, necesario tanto para

poder desarrollar aplicaciones como para poder ejecutarlas luego en entornos de prueba o producción.

## Variantes principales

*.Net Framework* Redistributable Package: este es el mínimo componente de la plataforma *.Net* que se necesita para poder ejecutar aplicaciones. Normalmente ésta es la variante que se instala en los entornos productivos, una vez que el desarrollo y las pruebas de la aplicación han finalizado.

Está compuesto por:

- El entorno de ejecución de la plataforma *.NET*
- Las bibliotecas de funcionalidad reutilizable
- *.NET Framework* SDK: esta versión contiene herramientas de desarrollo de línea de comandos (compiladores, depuradores), documentación de referencia, ejemplos y manuales para desarrolladores de aplicaciones.

Normalmente ésta variante se instala en los entornos de desarrollo de aplicaciones, y es más útil a los programadores que a los usuarios finales.

Para poder instalar la Versión SDK (*Software Development Kit*) es necesario instalar previamente el Redistributable Package.

- *NET Compact Framework*: esta es una versión reducida del *.NET Framework* Redistributable, especialmente pensada para ser instalada en dispositivos móviles como Pocket PC y Smartphone.

## 1.3.7 Sistema Gestor de base de datos

Un Sistema Gestor de Base de Datos (SGBD, en inglés DBMS: DataBase Management System) es un sistema de software que permite la definición de bases de datos; así como la elección de las estructuras de datos necesarios para el almacenamiento y búsqueda de los datos, ya sea de forma interactiva o a través de un lenguaje de programación. Un SGBD relacional es un modelo de datos que facilita a los usuarios describir los datos que serán almacenados en la base de datos junto con un grupo de operaciones para manejar los datos.

### Selección del Gestor de Base de datos

El gestor de Base de datos que se utilizará es PostgreSQL 9.0.3 pues es el sistema de gestión de bases de datos relacional orientada a objetos publicado bajo la licencia BSD más avanzado de todos. Mejora el rendimiento de las aplicaciones y hace el análisis de datos mucho más sencillo.

#### Ventajas:

- Restauración de bases de datos en procesos paralelos, que acelera recuperación de un respaldo hasta 8 veces.
- Privilegios por columna, que permiten un control más granular de datos confidenciales.
- Configuración de ordenamiento configurable por base de datos, lo cual hace a PostgreSQL más útil en entornos con múltiples idiomas.
- Nuevas herramientas de monitoreo de consultas que le otorgan a los administradores mayor información sobre la actividad del sistema.

### Conclusiones Parciales

En el presente capítulo se realizó una descripción de los conceptos básicos

## Sistema de Nóminas

---

---

relacionados con el dominio del problema a resolver. Se abordaron temas referentes a las nóminas y cómo se realizan, además de los sistemas de gestión de la información. Sé llevo un análisis sobre las herramientas a utilizar en el desarrollo del sistema y el porqué de su utilización.

## **CAPÍTULO 2: Características, análisis y diseño del sistema.**

### **Introducción**

En este capítulo para definir las características del sistema, se describen las principales funcionalidades de los sistemas de información, se realiza el modelado del dominio, se hace la propuesta del sistema describiendo como debe funcionar, se elaboran las historias de usuarios del negocio, las tareas de ingenierías y se incluye el diseño con metáforas y el diagrama de componentes.

### **2.1 Características y principales funcionalidades del sistema.**

El sistema a desarrollar tiene como principal objetivo gestionar toda la información referente a las nóminas del MININT en la Provincia de Artemisa. De esta forma se centraliza también en el cálculo de algunos elementos relacionados directamente con las nóminas

La gestión de la información de los datos consiste en Insertar, Modificar, Eliminar y Buscar datos referentes a las nóminas en el MININT. Cuando en dicha información se recoge los datos referentes a los trabajadores como son el nombre, los apellidos, la antigüedad como militar, el salario básico, así como los descuentos por créditos o por certificados médicos. Cuando se gestiona el comprobante de nómina todos los datos recogidos anteriormente se muestran y también se muestra el total a pagar que es el salario que se le va a pagar al trabajador después de pasar por las gestiones correspondientes para el cálculo del mismo.

El sistema cuenta con una interfaz principal que será asequible para cualquier

## Sistema de Nóminas

---

---

usuario, y cuenta con una interfaz de administrador solo asequible para los administradores del sistema garantizando que la gestión de los datos sea eficiente y segura. También brinda facilidad y rapidez en las tramitaciones de la información.

### 2.2 Planificación del proyecto por roles.

**Tabla #1**

<b>ROL</b>	<b>RESPONSABILIDAD</b>	<b>NOMBRE</b>
<b>Gerente (Manager)</b>	Dirige y controla las tareas del equipo. Toma las decisiones finales. Participa en la selección de objetivo y requerimientos. Controla el progreso y da seguimiento a cada iteración. Evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes.	Ariel Castro García
<b>Jefe de Proyecto (Manager)</b>	Organiza y guía las reuniones. Asegura condiciones adecuadas para el proyecto.	Ariel Castro García
<b>Cliente (Customer)</b>	Participa en las tareas que involucran la lista de reserva del producto.	Los trabajadores del área de economía del MININT



## Sistema de Nóminas

---



---

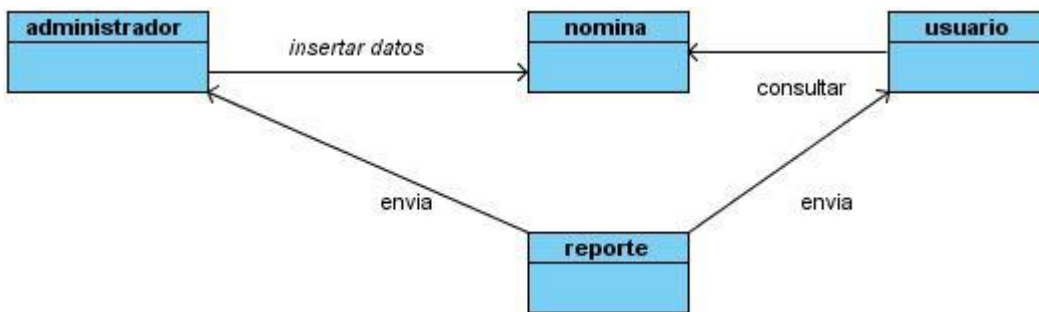
<b>Programadores (Programmers)</b>	<p>Elabora el código de las nuevas funcionalidades a implementar.</p> <p>Escribe las pruebas unitarias.</p> <p>Debe existir una comunicación y coordinación adecuada entre los programadores y el resto del equipo.</p>	Ariel Castro García
<b>Analista (Analyst)</b>	<p>Escribe las historias de usuario y las pruebas funcionales para validar su implementación.</p>	Ariel Castro García
<b>Diseñadores (Designers)</b>	<p>Encargados del diseño del sistema; así como el de los prototipos de interfaces, máximos responsables de la realización del diseño de las metáforas y supervisan el proceso de construcción.</p>	Ariel Castro García
<b>Encargado de Pruebas (Tester)</b>	<p>Es el encargado de ayudar al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el</p>	Ariel Castro García

	equipo y es responsable de las herramientas de soporte para pruebas.	
<b>Arquitecto (Architect)</b>	Se vincula directamente con el analista y el diseñador debido a que su trabajo tiene que ver con la estructura y el diseño en grande del sistema. Ayuda en el diseño de las metáforas.	Ariel Castro García

## 2.3 Modelo de dominio.

Dentro de las actividades más importantes definidas en la metodología SXP se encuentra la definición del Modelo de Historias de Usuarios del Negocio, en el cual se hace una detallada descripción del negocio. Si el negocio no se encuentra bien definido entre los clientes y los ejecutores del proyecto; entonces se genera el Modelo de Dominio.

A continuación se presenta el diagrama de dominio para el sistema que se propone.



**Figura # 1**

## 2.4 Lista de reserva del producto (LRP).

En la metodología SXP se describe otra actividad fundamental como lo es la Lista de Reserva del Producto (LRP), se recoge en ella una lista priorizada de todo el trabajo a realizar en el proyecto. Cuando un proyecto comienza es muy difícil tener claro todos los requerimientos sobre el producto, sin embargo, suelen surgir los más importantes que casi siempre son suficientes para una iteración.

Esta lista puede crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente. Con la restricción de que sólo puede cambiarse entre iteraciones. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto la lista debe acompañar los cambios en el entorno y el producto.

Esta lista puede estar conformada por requerimientos técnicos y del negocio, funciones, errores a reparar, defectos, mejoras y actualizaciones tecnológicas requeridas.

**Tabla # 2**

Prioridad	Ítem *	Descripción	Estimación	Estimado por
<b>Muy Alta</b>				
	1	Insertar información sobre los usuarios en el listado de nóminas.	2 días	Analista
	2	Modificar información sobre los usuarios en el listado de nóminas.	2 días	Analista
	3	Eliminar información sobre los usuarios en el listado de nóminas.	2 días	Analista

## Sistema de Nóminas

---

---

	4	Insertar información sobre los usuarios del Sistema de Nóminas	2 días	<i>Analista</i>
	5	Modificar información sobre los usuarios del Sistema de Nóminas.	2 días	<i>Analista</i>
	6	Eliminar información sobre los usuarios del Sistema de Nóminas.	2 días	<i>Analista</i>
	7	Insertar información sobre el submayor de crédito	2 días	<i>Analista</i>
	8	Modificar información sobre el submayor de crédito.	2 días	<i>Analista</i>
	9	Eliminar información sobre el submayor de crédito.	2 días	<i>Analista</i>
	10	Insertar información sobre el submayor de vacaciones.	2 días	<i>Analista</i>
	11	Modificar información sobre el submayor de vacaciones.	2 días	<i>Analista</i>
	12	Eliminar información sobre el submayor de vacaciones.	2 días	<i>Analista</i>

## Sistema de Nóminas

	13	Insertar información sobre el descuento de nóminas.	2 días	<i>Analista</i>
	14	Modificar información sobre el descuento de nóminas.	2 días	<i>Analista</i>
	15	Eliminar información sobre el descuento de nóminas.	2 días	<i>Analista</i>
	16	Insertar información sobre comprobante de nóminas.	2 días	<i>Analista</i>
	17	Modificar información sobre comprobante de nóminas.		
	18	Eliminar información sobre comprobante de nóminas.		
<b>Alta</b>				
	19	Buscar información sobre los usuarios en el listado de nóminas.	2 días	<i>Analista</i>
	20	Generar reporte sobre el listado de nóminas.	2 días	<i>Analista</i>
	21	Modificar reporte sobre el listado de nóminas.	2 días	<i>Analista</i>
	22	Buscar información sobre el submayor de	2 días	<i>Analista</i>

## Sistema de Nóminas

---

---

		crédito.		
	23	Generar reporte sobre el submayor de crédito.	2 días	<i>Analista</i>
	24	Modificar reporte sobre el submayor de crédito.	2 días	<i>Analista</i>
	25	Buscar información sobre el submayor de vacaciones.	2 días	<i>Analista</i>
	26	Generar reporte sobre el submayor de vacaciones.	2 días	<i>Analista</i>
	27	Modificar reporte sobre el submayor de vacaciones.	2 días	<i>Analista</i>
	28	Buscar información sobre el descuento de nóminas.	2 días	<i>Analista</i>
	30	Generar reporte sobre el descuento de nóminas.	2 días	<i>Analista</i>
	31	Modificar reporte sobre el descuento de nóminas.	2 días	<i>Analista</i>
	32	Buscar información sobre comprobante de nóminas.	2 días	<i>Analista</i>
	33	Generar reporte sobre el comprobante de	2 días	<i>Analista</i>

## Sistema de Nóminas

		nóminas.		
	34	Modificar reporte sobre el comprobante de nóminas.	2 días	Analista
<b>Media</b>				
<b>Baja</b>				
<b>Requisitos no funcionales</b>				
	35	Tener un ordenador con más de 1 GB de memoria RAM.		Analista
	36	Interfaz amigable.		Analista

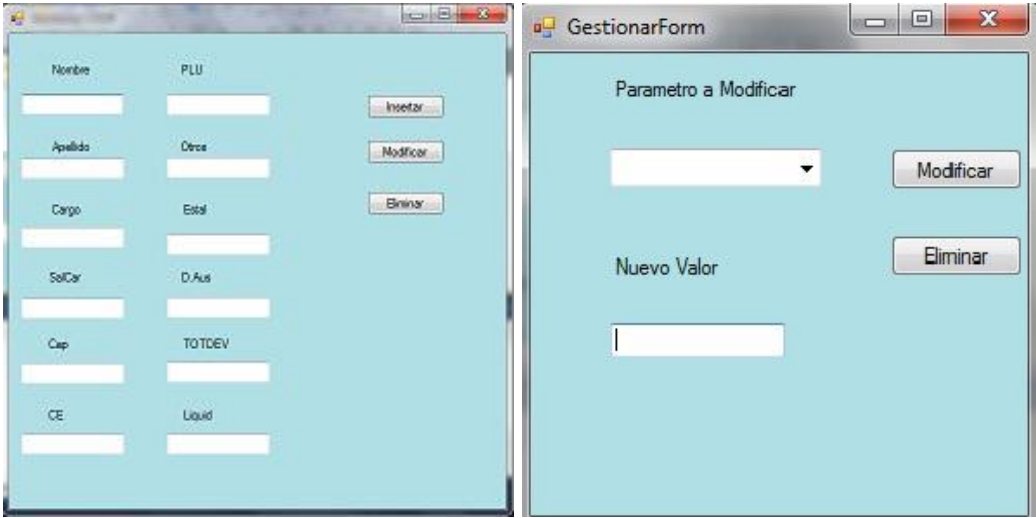
### 2.5 Historias de usuario y tareas de ingeniería.

En la metodología SXP se generan artefactos como las historias de usuarios, son las encargadas de describir las tareas que el sistema debe realizar, aspecto que depende de las especificaciones realizadas por el cliente.

**Tabla #4**

Historia de Usuario	
<b>Número:</b> HU_1	<b>Nombre Historia de Usuario:</b> Gestionar información sobre los empleados en el listado de nóminas
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> Ariel Castro García	<b>Iteración Asignada:</b> 2

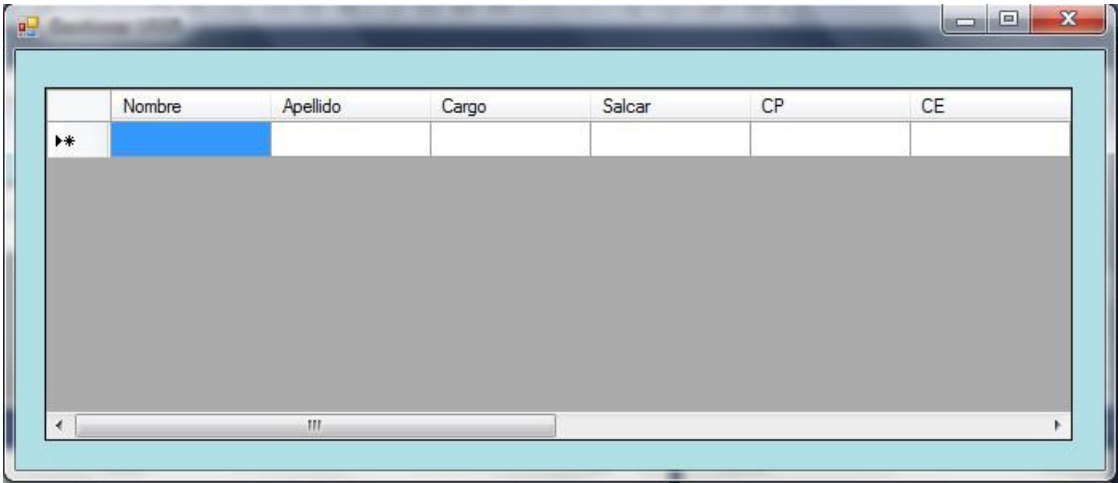
## Sistema de Nóminas

<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo insertar, modificar y eliminar toda la información referentes a los usuarios en el listado de nóminas	
<b>Observaciones:</b> Ninguna	
<b>Prototipo de interface:</b>	
	

<b>Historia de Usuario</b>	
<b>Número:</b> HU_2	<b>Nombre Historia de Usuario:</b> Generar reportes sobre los empleados en el listado de nóminas.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> Ariel Castro García	<b>Iteración Asignada:</b> 3

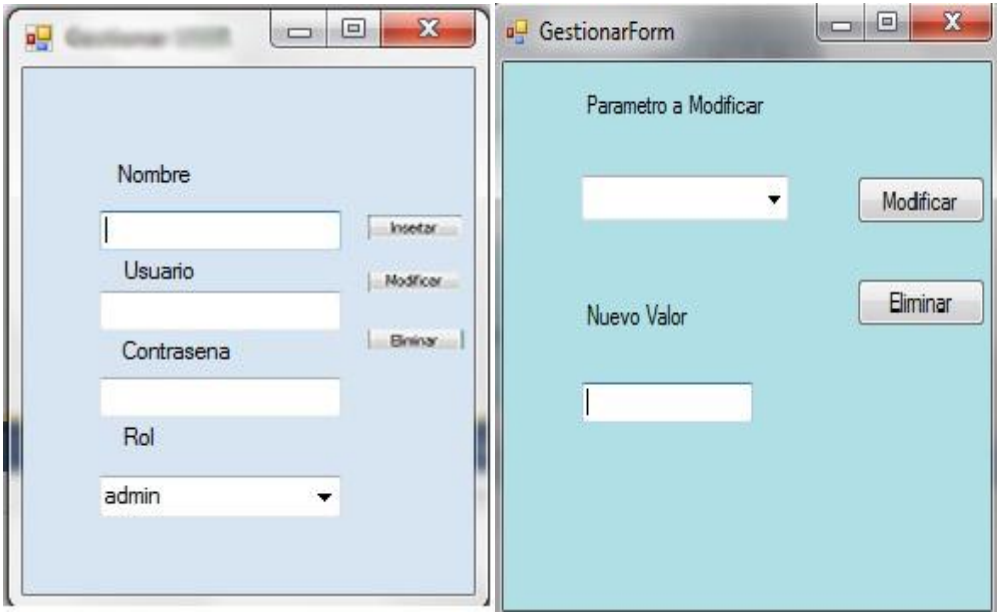


# Sistema de Nóminas

<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo generar los reportes de los usuarios en el listado de nóminas.	
<b>Observaciones:</b> Para que esto sea posible es necesario insertar los datos de los usuarios en el listado de nóminas.	
<b>Prototipo de interface:</b>	
	

<b>Historia de Usuario</b>	
<b>Número:</b> HU_3	<b>Nombre Historia de Usuario:</b>

# Sistema de Nóminas

Gestionar información de los usuarios del Sistema de Nóminas.	
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> Ariel Castro García	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo insertar, modificar y eliminar toda la información adquirida de los usuarios que van a trabajar con el Sistema de Nóminas	
<b>Observaciones:</b> Ninguna	
<b>Prototipo de interface:</b> 	

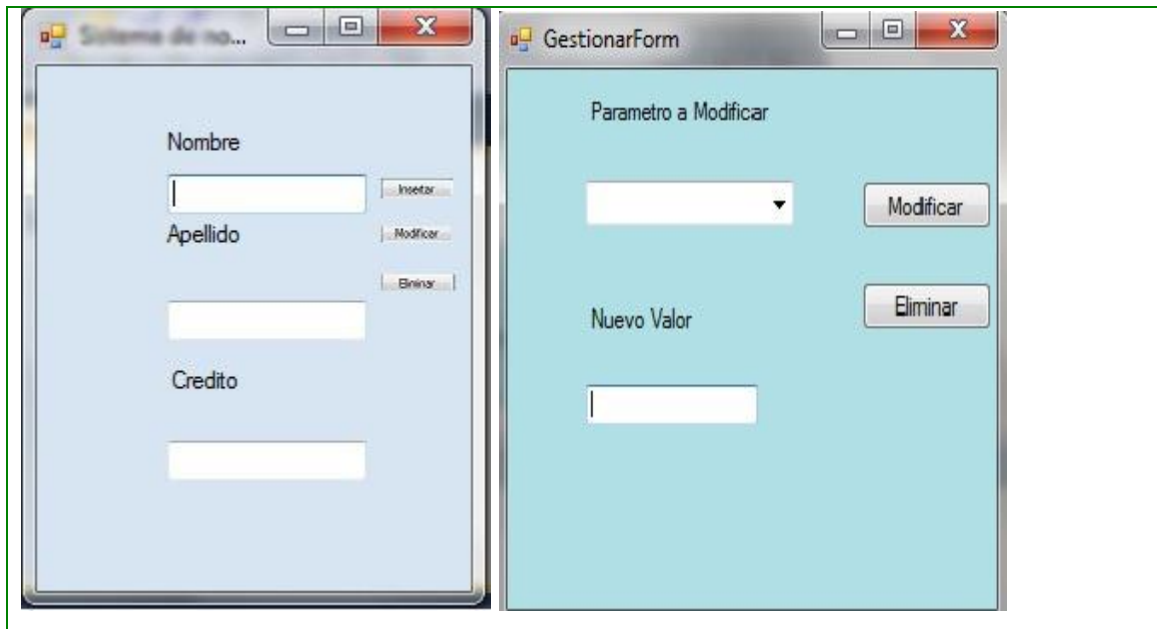
## Sistema de Nóminas

---

---

Historia de Usuario	
<b>Número:</b> HU_4	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el submayor de créditos.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> Ariel Castro García	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo insertar y modificar toda la información referente al submayor de créditos.	
<b>Observaciones:</b>	
<b>Prototipo de interface:</b>	

# Sistema de Nóminas

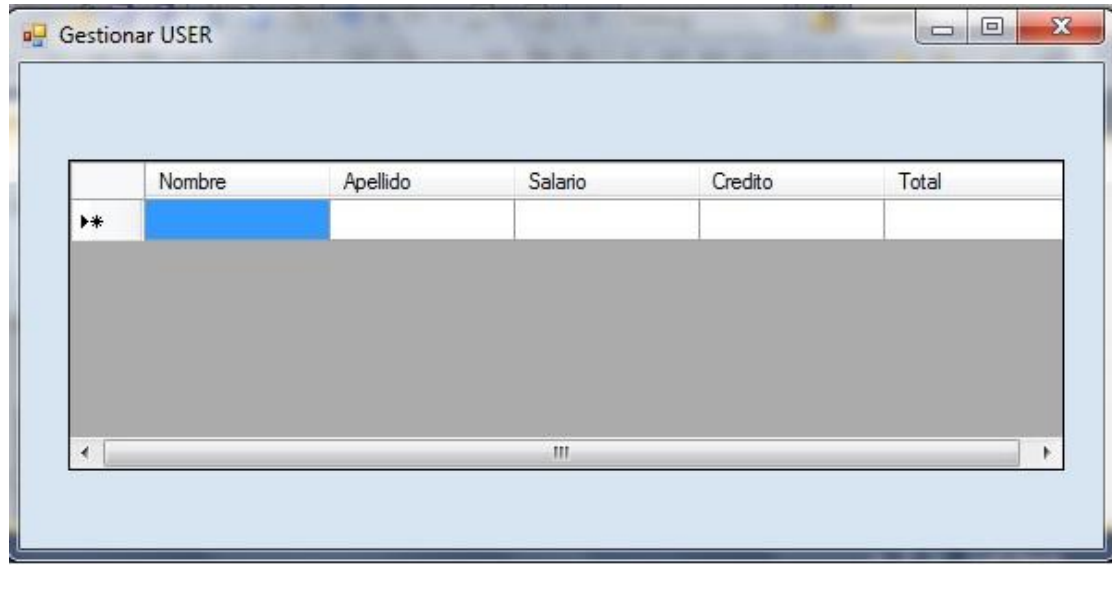


Historia de Usuario	
<b>Número:</b> HU_5	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el submayor de créditos.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> Ariel Castro García	<b>Iteración Asignada:</b> 3
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Observaciones:</b>	

# Sistema de Nóminas

Para que esto sea posible es necesario insertar los datos requeridos por el submayor de créditos.

## Prototipo de interface:



Historia de Usuario	
<b>Número:</b> HU_6	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el submayor de vacaciones.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ▲ Ariel Castro García	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1

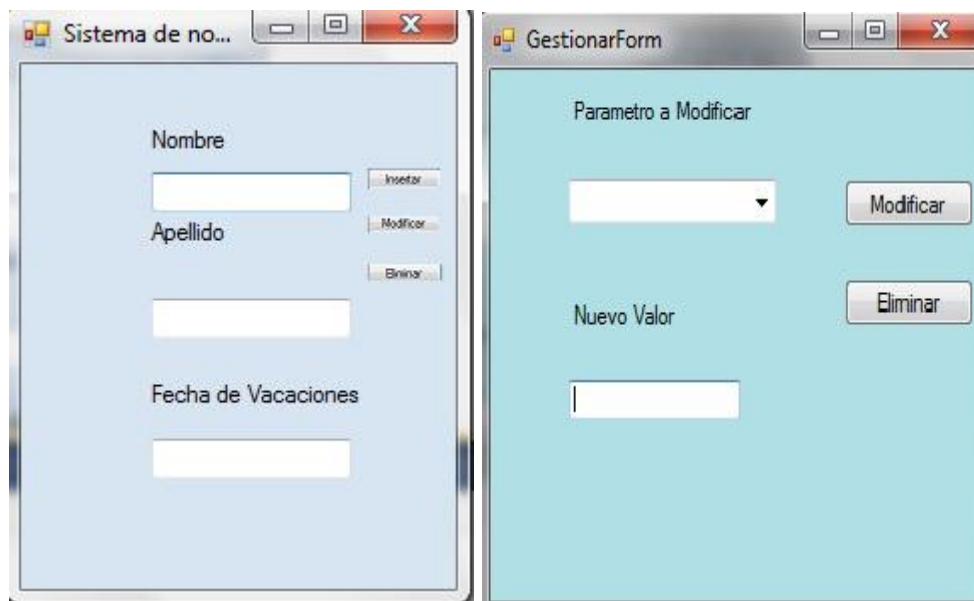
# Sistema de Nóminas

## Descripción:

La presente historia de usuario tiene como objetivo insertar, modificar, eliminar y buscar toda la información referente al submayor de vacaciones.

**Observaciones:** Ninguna

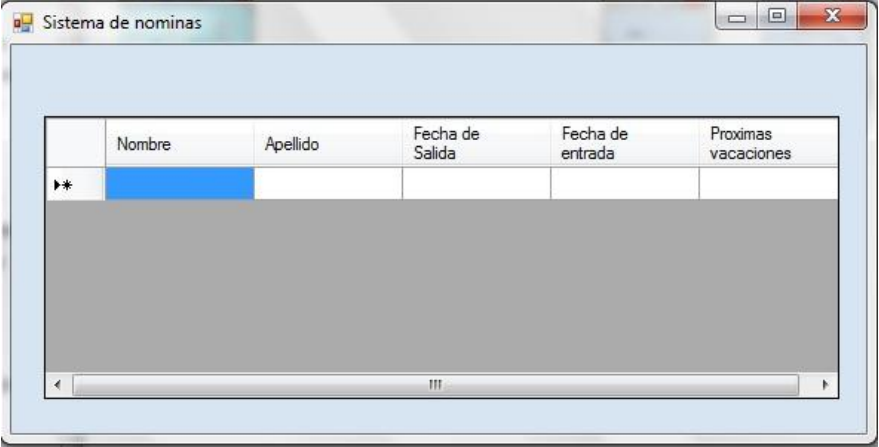
## Prototipo de interface:



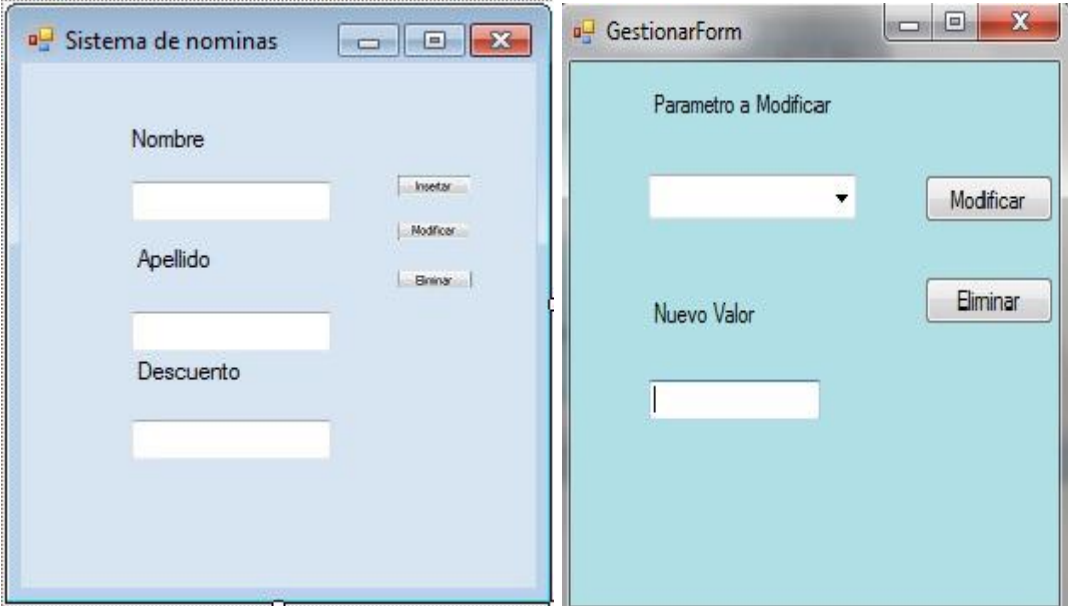
## Historia de Usuario

<b>Número:</b> HU_7	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el submayor de vacaciones.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ▲ Ariel Castro García	<b>Iteración Asignada:</b> 3

## Sistema de Nóminas

<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo generar los reportes del submayor de vacaciones.	
<b>Observaciones:</b> Para que esto sea posible es necesario insertar los datos requeridos por el submayor de vacaciones.	
<b>Prototipo de interface:</b>	
	
<b>Historia de Usuario</b>	
<b>Número:</b> HU_8	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el descuento de nóminas
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ▲ Ariel Castro García	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1

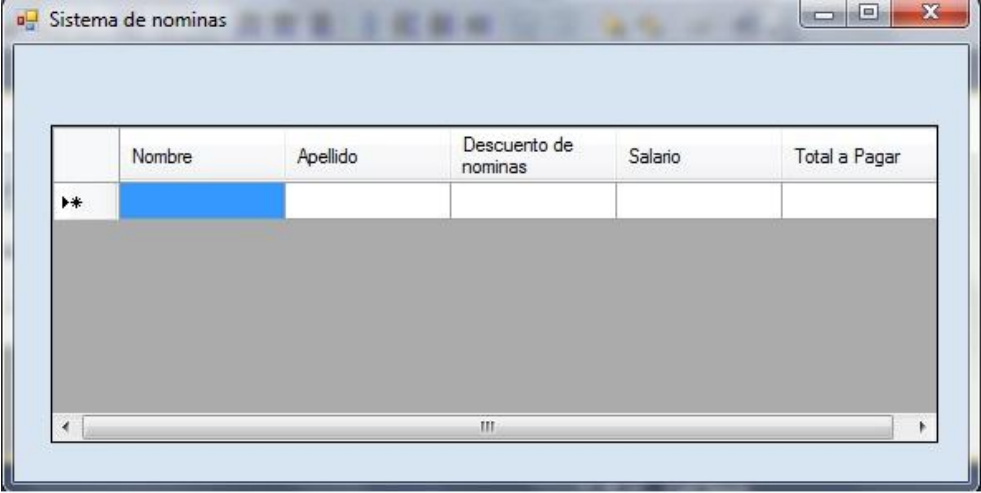
# Sistema de Nóminas

<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo insertar, modificar, eliminar y buscar toda la información referente a todos los descuentos de nóminas.	
<b>Observaciones:</b> Ninguna	
<b>Prototipo de interface:</b> 	

Historia de Usuario	
<b>Número:</b> HU_9	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el descuento de nóminas.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ▲ Ariel Castro García	<b>Iteración Asignada:</b> 3

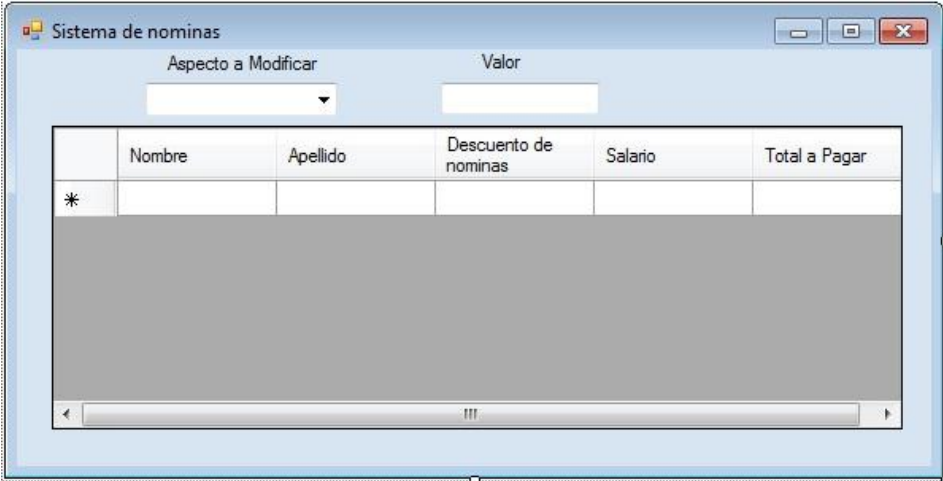


# Sistema de Nóminas

<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo generar los reportes referentes a todos los descuentos de nóminas poseído por un determinado usuario del Sistema de Nóminas.	
<b>Observaciones:</b> Para que sea posible esta acción es necesario introducir los datos referentes a los descuentos de nóminas	
<b>Prototipo de interface:</b>	
	

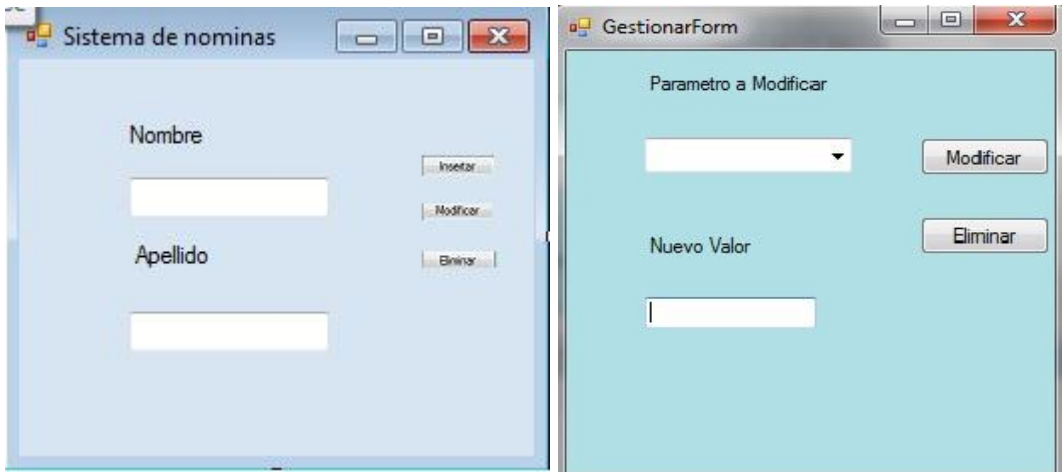
Historia de Usuario	
<b>Número:</b> HU_10	<b>Nombre Historia de Usuario:</b> Modificar reportes sobre descuento de nóminas.

# Sistema de Nóminas

<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ^ Ariel Castro García	<b>Iteración Asignada:</b> 4
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo modificar los reportes referentes a todos los descuentos de nóminas que posea cada usuario	
<b>Observaciones:</b> Para realizar esta acción es necesario señalar el usuario a modificar seguidamente se cargará toda la información de éste y se pasará a introducir los datos con los que se quiere efectuar el cambio.	
<b>Prototipo de interface:</b>	
	

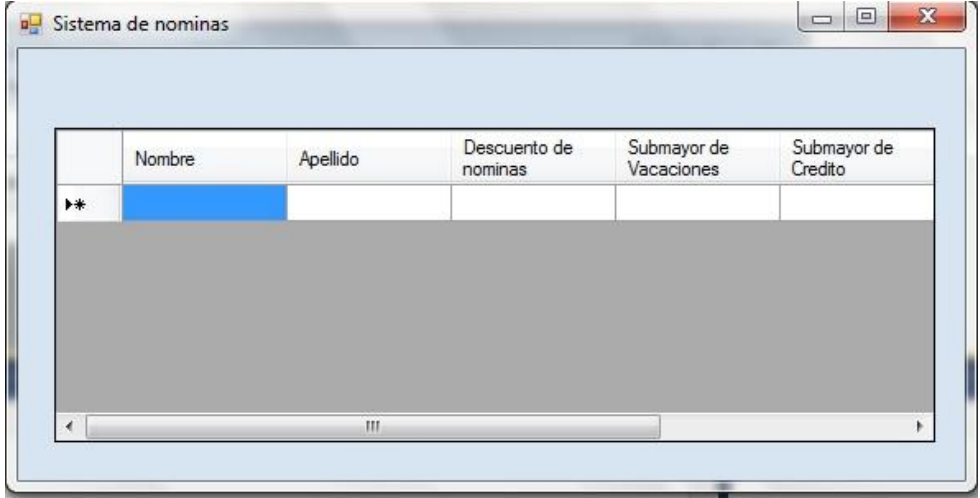
**Historia de Usuario**

## Sistema de Nóminas

<b>Número:</b> HU_11	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el comprobante de nóminas.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ▲ Ariel Castro García	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo insertar, modificar, eliminar y buscar toda la información referente a los comprobantes de nóminas.	
<b>Observaciones:</b> Ninguna	
<b>Prototipo de interface:</b>	
	

Historia de Usuario	
<b>Número:</b> HU_12	<b>Nombre Historia de Usuario:</b>

# Sistema de Nóminas

	Generar reportes sobre el comprobante de nóminas.
<b>Modificación de Historia de Usuario Número:</b> ninguna	
<b>Usuario:</b> ✦ Ariel Castro García	<b>Iteración Asignada:</b> 3
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo generar los reportes referentes al comprobante de nóminas.	
<b>Observaciones:</b> Para que sea posible esta acción es necesario introducir los datos referentes al comprobante	
<b>Prototipo de interface:</b>	
	

## **2.6 Plan de release.**

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
<i>Iteración 2</i>	En esta iteración se van a realizar la historia de usuario gestionar.	HU_1, HU_4, HU_5, HU_8, HU_11, HU_14.	<i>5 semanas</i>
Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
<i>Iteración 3</i>	En esta iteración se van a realizar la historia de usuario generar.	HU_2, HU_6, HU_9, HU_12, HU_15, H	<i>5 semanas</i>
Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
<i>Iteración 4</i>	En esta iteración se van a realizar la historia de usuario modificar.	HU_3, HU_7, HU_10, HU_13, HU_16,	<i>5 semanas</i>

## **2.7 Arquitectura de software.**

La arquitectura de software representa la columna vertebral de todo software y es el diseño de más alto nivel de la estructura de un sistema. Ocurre muy tempranamente en el ciclo de vida y define los estilos o grupos de estilos adecuados para cumplir con los requerimientos no funcionales.

# Sistema de Nóminas

---

---

Se utilizará una arquitectura de 3 capas o niveles cuyo objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario. A continuación se enumeran las capas:

- **Capa de presentación:** es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información.
- **Capa de negocio:** contiene los objetos y servicios de negocio de la aplicación. Recibe peticiones de la capa de presentación y procesa la lógica de negocio basada en las peticiones.
- **Capa de datos:** es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

El estilo arquitectónico a utilizar es el estilo llamada y retorno, con el patrón de arquitectura de software Modelo-Vista-Controlador. (MVC), dicho patrón se centra en una arquitectura de 3 capas.

## Ventajas

Una separación total entre lógica de negocio y presentación. A esto se le pueden aplicar opciones como el multilenguaje, distintos diseños de presentación, sin alterar la lógica de negocio. La separación de capas como presentación, lógica de negocio, acceso, lo que al final resulta en un ahorro de tiempo en desarrollo en posteriores proyectos

- Agregar nuevas formas de recolectar las ordenes del usuario (interpretar sus modelos mentales)
- Modificar los objetos de negocios bien sea para mejorar el performance o para

migrar a otra tecnología.

- Las labores de mantenimiento también se simplifican y se reduce el tiempo necesario para ellas. Las correcciones solo se deben hacer en un solo lugar y no en varios como sucedería si se tuviese una mezcla de presentación e implementación de la lógica del negocio .
- Las vistas también son susceptibles de modificación sin necesidad de provocar que todo el sistema se paralice. Adicionalmente el patrón MVC propende a la especialización de cada rol del equipo, por tanto en cada liberación de una nueva versión se verán los resultados.

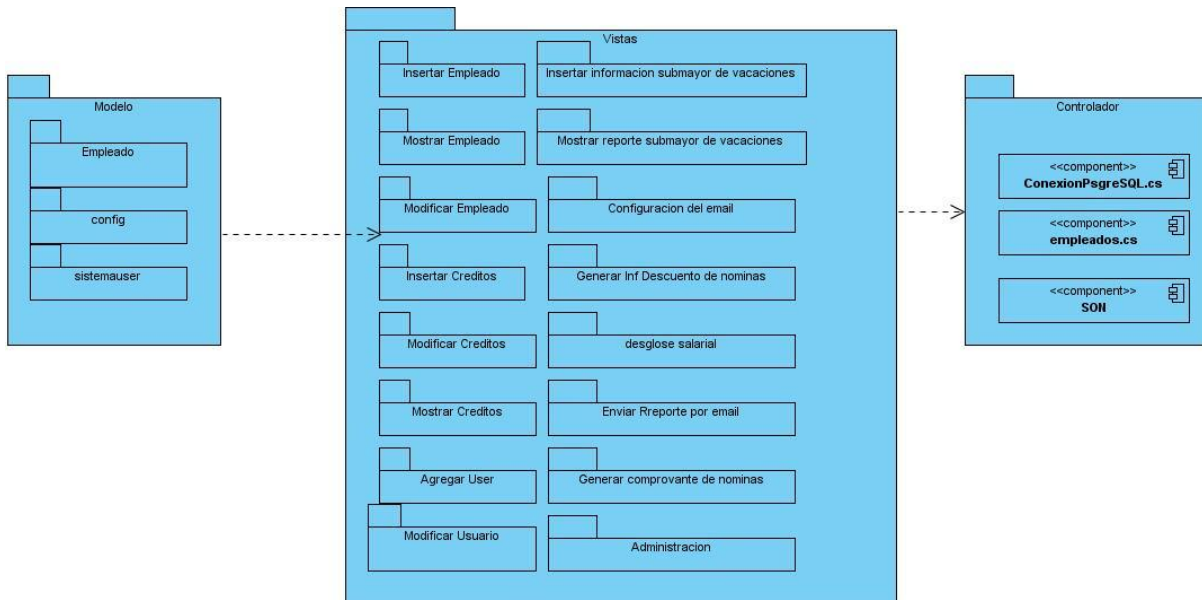
### **2.8 Diseño con metáforas.**

Debido a que SXP está basada en XP, y dicha metodología define un término llamado metáfora, lo cual según Martin Fowler es una historia compartida que describe como debería funcionar el sistema y define que la práctica de la metáfora consiste en formar un conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema.

El Diseño con metáforas es sencillamente el diseño de la solución más simple que pueda funcionar y ser implementado en un momento dado del proyecto; lo cual genera el artefacto conocido como Modelo de Diseño, que a su vez está compuesto por un diagrama de paquetes, el cual expone dicho diseño. Los diagramas de paquetes describen los elementos físicos del sistema y sus relaciones. Muestra las dependencias de compilación de los ficheros de código, relaciones de derivación entre ficheros de código fuente y ficheros que son resultados de la compilación, dependencias entre elementos de implementación y los correspondientes elementos de diseños que son implementados. Estos muestran además la organización y las dependencias lógicas entre un conjunto de componentes

# Sistema de Nóminas

software, sean éstos componentes de código fuente, librerías, binarios o ejecutables.



## **Descripción:**

En la capa de **vistas** se tiene los formularios los se utilizan para crear la interfaz visual principal de la aplicación Sistema de Nóminas del MININT (SON). Esta capa depende de la capa **Controlador** para su funcionamiento.

En la capa **Controlador** se integran todas las clases del sistema.

En la capa **Modelo** se tienen las tablas que integran la base de datos.

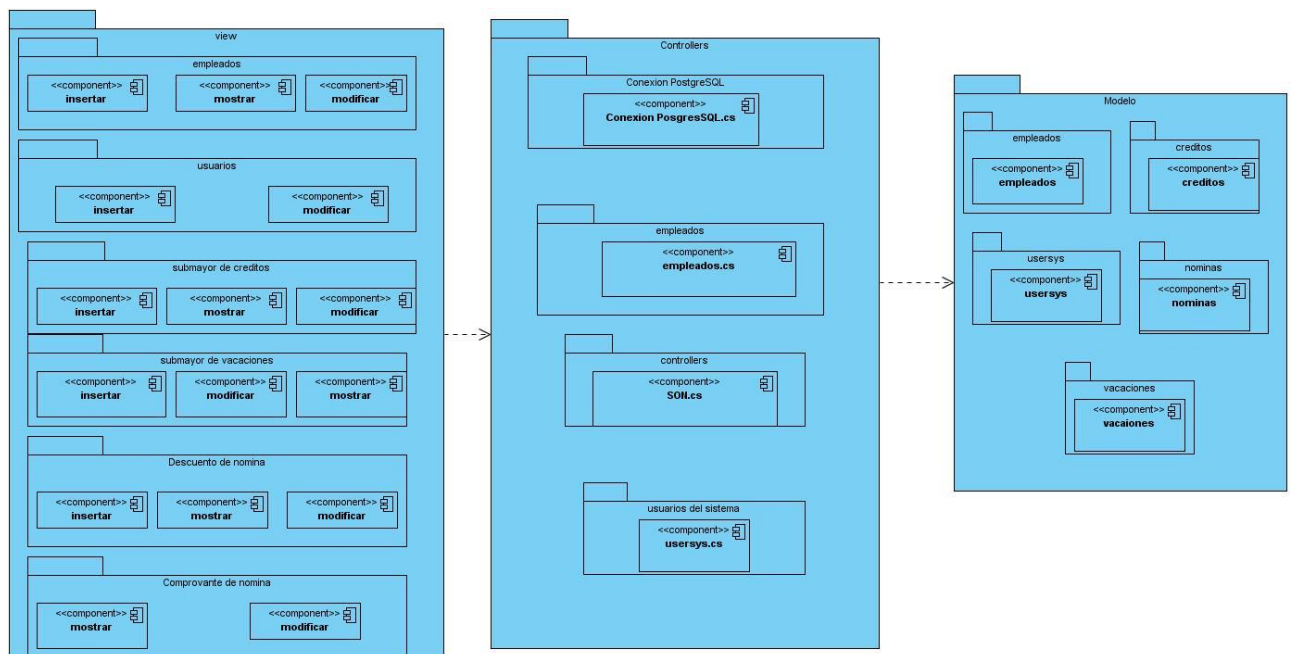
## **2.9 Diagrama de componentes.**

Un **diagrama de componentes** es un diagrama tipo del Lenguaje Unificado de Modelado.



# Sistema de Nóminas

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes. Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema.



## Conclusiones parciales

En el capítulo se definen las características de los sistemas, así como las principales funcionalidades del *software*, se aborda acerca de las características de los sistemas de información y se realiza el modelo del dominio así como diagramas de paquetes y componentes.

## CAPÍTULO 3. Validación implementación.

### Introducción

En este capítulo se presentan los casos de pruebas de aceptación que se le

realizaron al sistema, las cuales son test que se le hacen a la aplicación para probar su funcionamiento. En este capítulo también se dan a conocer los resultados obtenidos hasta el momento.

### 3.1 Casos de pruebas

La programación extrema define entre iteración e iteración un conjunto de casos de pruebas de aceptación para poder avanzar a una iteración superior. Durante el desarrollo del Sistema nóminas para las oficinas de economía y finanzas del MININT se diseñaron un conjunto de casos de prueba a las que fue sometido el sistema para comprobar el funcionamiento de acuerdo a las Historias de Usuario. Se definieron casos de prueba para todas las historias de usuario, las cuales se presentan a continuación.

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> <b>SON011</b>	<b>Nombre Historia de Usuario:</b> Gestionar información sobre los empleados en el listado de nóminas
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se introdujeron los datos de los empleados del Sistema de Nóminas</i>	
<b>Condiciones de Ejecución:</b> <i>Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se introduce el nombre , los apellidos y los datos relacionados con el salario del empleado y a continuación se probara a insertarlos en la base de datos</i>	

## Sistema de Nóminas

---

---

**Resultado Esperado:**

- Se insertaron los datos correctamente en la base de datos.

**Evaluación de la Prueba:** *Satisfactoria*

### Caso de Prueba de Aceptación

**Código Caso de Prueba:**  
**SON012**

**Nombre Historia de Usuario:** Gestionar información sobre los empleados en el listado de nóminas

**Nombre de la persona que realiza la prueba:** *Ariel Castro García*

**Descripción de la Prueba:** *En la siguiente prueba se modificó los datos de los empleados del Sistema de Nóminas.*

**Condiciones de Ejecución:** *Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos*

**Entrada / Pasos de ejecución:** *Se obtiene el nombre , los apellidos y los datos relacionados con el salario del empleado que se quiere modificar y a continuación se probara a modificarlos en la base de datos*

**Resultado Esperado:**

- Se modificaron los datos correctamente en la base de datos.

**Evaluación de la Prueba:** *Satisfactoria*

### Caso de Prueba de Aceptación

## Sistema de Nóminas

---

---

<b>Código Caso de Prueba:</b> SON022	<b>Nombre Historia de Usuario:</b> Gestionar configuración de la Base de Datos.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba configura la base de datos con la que trabaja el sistema.</i>	
<b>Condiciones de Ejecución:</b> <i>no posee condiciones</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se introduce el número IP o nombre de la PC donde estará almacenada la base de datos y el nombre el usuario y la contraseña de la base de datos.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• La base de datos se configuro correctamente.</li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

### Caso de Prueba de Aceptación

<b>Código Caso de Prueba:</b> SON033	<b>Nombre Historia de Usuario:</b> Registrarse en el sistema.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se probara a registrar un usuario en el sistema.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos.</i>	

## Sistema de Nóminas

---

---

<b>Entrada / Pasos de ejecución:</b> <i>Se introducirá el usuario y la contraseña</i>
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• El usuario se registro correctamente.</li></ul>
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON044	<b>Nombre Historia de Usuario:</b> Gestionar información de los usuarios del sistema.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se probara a crear un usuario en el sistema.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se llenaran los datos del usuario para registrarlo en el sistema.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• Se creó un usuario administrador del sistema.</li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON055	<b>Nombre Historia de Usuario:</b> Gestionar información de los usuarios del sistema.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se probará a crear un usuario en el sistema.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se llenarán los datos del usuario para registrarlo en el sistema.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• Se creó un usuario administrador del sistema.</li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON066	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el submayor de créditos.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	

## Sistema de Nóminas

---

---

<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se insertara la información de un empleado referente al crédito.</i>
<b>Condiciones de Ejecución:</b> <i>Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos</i>
<b>Entrada / Pasos de ejecución:</b> <i>Se llenaran los campos nombre, apellidos y créditos.</i>
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• Se inserto en la base de datos la información del submayor de crédito.</li></ul>
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON066	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el submayor de créditos.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>En la siguiente prueba se modificara la información de un empleado referente al crédito.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe estar registrado como administrador del sistema y haber llenado el registro de configuración de la base de datos</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se llenaran los campos nombre, apellidos y créditos.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• Se modifiko en la base de datos la información del submayor de</li></ul>	

## Sistema de Nóminas

---

---

crédito.

**Evaluación de la Prueba:** *Satisfactoria*

### Caso de Prueba de Aceptación

**Código Caso de Prueba:**  
SON077

**Nombre Historia de Usuario:** Generar reportes sobre el submayor de créditos.

**Nombre de la persona que realiza la prueba:** *Ariel Castro García*

**Descripción de la Prueba:** *En la siguiente prueba se presiona el botón mostrar que deberá mostrar toda la información referente al crédito.*

**Condiciones de Ejecución:** *Debe haber llenado el registro de configuración de la base de datos.*

**Entrada / Pasos de ejecución:** *Se presiona el botón mostrar.*

**Resultado Esperado:**

- *Se mostró toda la información referente al crédito.*

**Evaluación de la Prueba:** *Satisfactoria*

### Caso de Prueba de Aceptación

**Código Caso de Prueba:**

**Nombre Historia de Usuario:** Gestionar información sobre el submayor de vacaciones.



## Sistema de Nóminas

---

SON088	
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se probara a calcular el salario de las vacaciones de un empleado.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos y estar registrado como administrador del sistema.</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se introduce el nombre los apellidos y el tiempo que estará de vacaciones el empleado</i>	
<p style="text-align: center;"><b>Resultado Esperado:</b></p> <ul style="list-style-type: none"> <li>• <i>El salario se genero satisfactoriamente</i></li> </ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON088	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el submayor de vacaciones.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se probara a insertar en la base de datos los datos de las vacaciones de un empleado.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos y estar registrado como administrador del sistema.</i>	

## Sistema de Nóminas

---

---

<b>Entrada / Pasos de ejecución:</b> <i>Se introduce el nombre los apellidos y el tiempo que estará de vacaciones el empleado</i>
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• <i>Las vacaciones se insertaron satisfactoriamente.</i></li></ul>
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON099	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el submayor de vacaciones.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se muestran todos los datos referentes al submayor de vacaciones.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos y estar registrado como administrador del sistema.</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se ejecutara el botón mostrar el cual debe devolver los datos sobre el submayor de vacaciones.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• <i>Se mostrara los datos acerca el submayor de vacaciones.</i></li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

## Sistema de Nóminas

---

---

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON1010	<b>Nombre Historia de Usuario:</b> Gestionar información sobre el descuento de nóminas.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se gestionaran todos los datos referentes al descuento de nómina (ausencias y motivos).</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos y estar registrado como administrador del sistema.</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se insertaran y se modificaran los datos que puedan ocasionar un descuento en la nómina de un empleado.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• <i>Se insertaran en la base de datos los datos referentes al descuento de nóminas.</i></li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON1111	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el descuento de nóminas.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	

## Sistema de Nóminas

---

---

<b>Descripción de la Prueba:</b> <i>Se deberá generar un reporte sobre el reporte de nóminas.</i>
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos.</i>
<b>Entrada / Pasos de ejecución:</b> <i>Cuando se insertan los datos el reporte aparece automáticamente</i>
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• Se mostraron todos los datos asociados al descuento de la nómina.</li></ul>
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON1212	<b>Nombre Historia de Usuario:</b> Insertar configuración de correo electrónico.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se probara a insertar la configuración del correo electrónico.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos, debe estar registrado como administrador del sistema y debe haber entrado una cuenta de correo electrónico con su contraseña validas.</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se insertan el servidor el puerto y la</i>	

## Sistema de Nóminas

---

---

*utilización del protocolo SSL.*

**Resultado Esperado:**

- *Se inserto correctamente la configuración.*

**Evaluación de la Prueba:** *Satisfactoria.*

### Caso de Prueba de Aceptación

**Código Caso de Prueba:**  
SON1213

**Nombre Historia de Usuario:** Envió de reporte por correo.

**Nombre de la persona que realiza la prueba:** *Ariel Castro García*

**Descripción de la Prueba:** *Se probara que el sistema envíe el reporte de la nómina por correo electrónico.*

**Condiciones de Ejecución:** *Debe haber llenado el registro de configuración de la base de datos, debe estar registrado como administrador del sistema y debe haber entrado una cuenta de correo electrónico con su contraseña validas.*

**Entrada / Pasos de ejecución:** *Se insertan el nombre y los apellidos del empleado del cual se desea que se envíe el reporte de la nómina por correo.*

**Resultado Esperado:**

- *Se envió un reporte por correo electrónico.*

**Evaluación de la Prueba:** *Satisfactoria*

## Sistema de Nóminas

---

---

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON1313	<b>Nombre Historia de Usuario:</b> Generar reportes sobre el comprobante de nóminas.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se probara que el sistema genere toda la información de la nómina.</i>	
<b>Condiciones de Ejecución:</b> <i>Debe haber llenado el registro de configuración de la base de datos.</i>	
<b>Entrada / Pasos de ejecución:</b> <i>Se insertan el nombre y los apellidos del empleado del cual se desea generar un reporte de nómina.</i>	
<b>Resultado Esperado:</b> <ul style="list-style-type: none"><li>• <i>Se genero el reporte de la nómina</i></li></ul>	
<b>Evaluación de la Prueba:</b> <i>Satisfactoria</i>	

Caso de Prueba de Aceptación	
<b>Código Caso de Prueba:</b> SON1414	<b>Nombre Historia de Usuario:</b> Administrar sistema.
<b>Nombre de la persona que realiza la prueba:</b> <i>Ariel Castro García</i>	
<b>Descripción de la Prueba:</b> <i>Se probara que la ventana de administración</i>	

*tenga los vínculos a las secciones que el administrador debe acceder*

**Condiciones de Ejecución:** *Debe haber llenado el registro de configuración de la base de datos y estar registrado como usuario administrador del sistema.*

**Entrada / Pasos de ejecución:** *Se probara que los botones de las distintas secciones funcionen.*

**Resultado Esperado:**

- *Cada botón lleva a la sección correspondiente*

**Evaluación de la Prueba:** *Satisfactoria*

### 3.2 Casos de pruebas unitarias.

Los casos de pruebas unitarias son procesos que se le realizan a los componentes individuales de un programa. Su mayor propósito es encontrar las diferencias entre la especificación de la interfaz del modulo y comportamiento real. La importancia de esta fase es vital en sistemas de tiempo real u otros en los que los errores sean irrecuperables. Las pruebas no tienen el objeto de prevenir errores sino de detectarlos. Se efectúan sobre el trabajo realizado y se deben encarar con la intención de descubrir la mayor cantidad de errores posible. Existen diversos tipos de pruebas de software como lo son: **Prueba de caja blanca:** Cuando una aplicación es probada desde adentro, usando la lógica aplicativa. Las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo
- Se utilizan las decisiones en su parte verdadera y en su parte falsa
- Se ejecuten todos los bucles en sus límites
- Se utilizan todas las estructuras de datos internas.

Las pruebas de caja Blanca realizadas al Sistema de Nóminas, se realizaron con la herramienta UNIT TEST, herramienta que posee el IDE Visual Studio para la realización de pruebas unitarias.

Anexo #7 y #8

**Prueba de caja negra:** cuando una aplicación es probada usando su interfaz externa. Una prueba de tipo Caja Negra se lleva a cabo sin tener conocimiento de la estructura/funcionamiento interno del sistema.

En las pruebas de caja negra realizadas al sistema de nómina se probó cada interfaz del sistema dando todas resultados positivos.

Anexo #4.

### **3.3 Resultados y funcionalidades obtenidas.**

Como resultado de este trabajo queda disponible la versión 1.0 del Sistema de Nóminas.

Entre las principales funcionales que posee el Sistema de Nóminas se pueden mencionar:

Almacena y muestra los empleados de un entidad.

Almacena y muestra del crédito de los empleados.

Calculo del salario de las vacaciones.

Muestra el desglose salarial.

Calculo del salario total.

Envío de reporte de nóminas por correo electrónico.

### **3.4 Aporte social y económico.**

La creación de un software que logre automatizar el proceso de nóminas, reduce en gran cantidad el tiempo en que se realizan las nóminas en un centro. El tiempo en que una persona realiza los cálculos se ve reducido a más de un 50 % , también



## Sistema de Nóminas

---

---

se ven reducidas las posibilidades de un error de cálculo en la creación del documento , así como la funcionalidad del software de enviar reportes por email le da la posibilidad a cada trabajador del centro de conocer el estado de su nómina.

### **Conclusiones**

Al concluir el desarrollo de las pruebas al sistema se demuestra con el análisis de los resultados obtenidos, que las funcionalidades alcanzadas por el Sistema de Nóminas, se han desarrollado según los requerimientos definidos en la etapa inicial y en el período establecido. Funcionalidades necesarias para cumplir con las necesidades del organismo a quien va destinado el software.

## **Conclusiones Generales**

Con el fin de desarrollar el sistema de nóminas para las oficinas de economía y finanzas de MININT de la Provincia de Artemisa, se elaboró la fundamentación teórica de la investigación utilizando métodos de investigación teóricos y empíricos. Se realizó una amplia revisión bibliográfica y se determinó el estado del arte, lo que permitió realizar una correcta selección de las herramientas y metodologías idóneas para el desarrollo de la aplicación. Mediante la utilización del lenguaje de programación y de .NET frameworks se logró que la implementación del sistema culminara en el tiempo planificado y con la calidad requerida, respondiendo satisfactoriamente a las necesidades del cliente. El uso de la metodología SXP permitió la introducción de procedimientos ágiles que permitieron actualizar los procesos de software de forma rápida y eficiente beneficiando la actividad productiva. Por último se aplicaron pruebas de aceptación, de caja Blanca y caja negra para validar la calidad del resultado obtenido, las cuales fueron evaluadas de satisfactorias en su totalidad.

## **Recomendaciones**

Se recomienda para futuras versiones de este software agregarle funcionalidades tales como exportar nomina, imprimir nomina. Para la utilización de este software se debe tener en cuenta que esta es específicamente para el proceso de nómina que se realiza en la provincia de Artemisa.

## Glosario de términos

**Framework:** término usado en programación orientada a objetos para definir un conjunto de clases que definen un diseño abstracto para solucionar un conjunto de problemas relacionados.

**Historia de usuario:** término en el cual se resume todo lo referente a las especificaciones de los requerimientos, así como lo relacionado con los casos de uso, conocidos del proceso unificado.

**SQL:** es un lenguaje de base de datos normalizado, utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre los datos o sobre la estructura de los mismos. Pero como sucede con cualquier sistema de normalización hay excepciones para casi todo; de hecho, cada motor de bases de datos tiene sus peculiaridades y lo hace diferente de otro motor, por lo tanto, el lenguaje SQL normalizado (ANSI) no servirá para resolver todos los problemas, aunque si se puede asegurar que cualquier sentencia escrita en ANSI será interpretable por cualquier motor de datos.

**C# :**(pronunciado si sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma.NET, que después fue aprobado como un estándar por la ECMA e ISO.

## REFERENCIAS BIBLIOGRÁFICAS

Rena, F. Manual de Programación en C para principiantes y avanzados

REMENTERIA PIÑONES, A. Concepto y alcance de Gestión. 2002.

MARTÍNEZ, A. Alternativa metodológica que propicie un aprendizaje desarrollador en los alumnos del colegio de bachilleres en el área histórica social. (2000), Gro. 120 h. Tesis (en opción al título académico de Máster en Investigación Educativa) Ministerio de Educación.

DÍAZ PÉREZ, M., DE LIZ CONTRERAS, Y., RIVERO AMADOR, S., Máster en Bibliotecología y Ciencias de la Información, “Características de los sistemas de información que permiten la gestión oportuna de la información y el conocimiento constitucional”.

Disponible en:

[http://scielo.sld.cu/scielo.php?pid=S102494352009001100006&script=sci\\_arttext](http://scielo.sld.cu/scielo.php?pid=S102494352009001100006&script=sci_arttext)

WOODMAN, L. Information management in large organizations. En: Information management from strategies to action. London:ASLIB, 1985, págs. 95-97.

Pérez Rodríguez Y, Coutín Domínguez A. La gestión del conocimiento: un nuevo enfoque en la gestión empresarial. Acimed 2005; 13(6). Disponible en: [http://bvs.sld.cu/revistas/aci/vol13\\_6\\_05/aci04605.htm](http://bvs.sld.cu/revistas/aci/vol13_6_05/aci04605.htm)

ANDREU, R.; RICART, J.E.; VALOR, J. Estrategia y Sistemas de Información. McGraw Hill. 1996.

## Anexos

Imágenes del software.

Anexo # 1 Ventana principal del SON

El crédito ha sido modificado

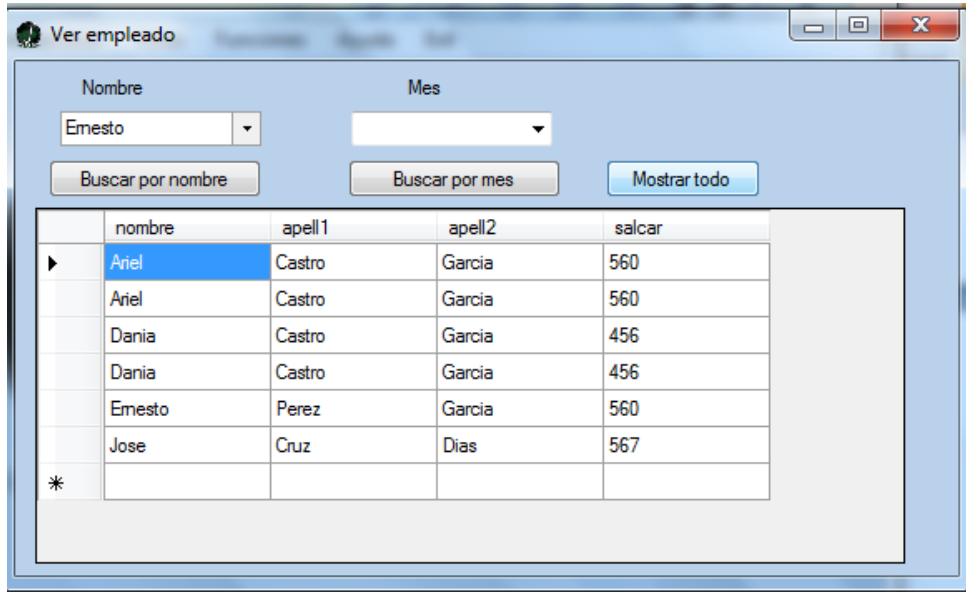


Anexo #2 Panel de administración

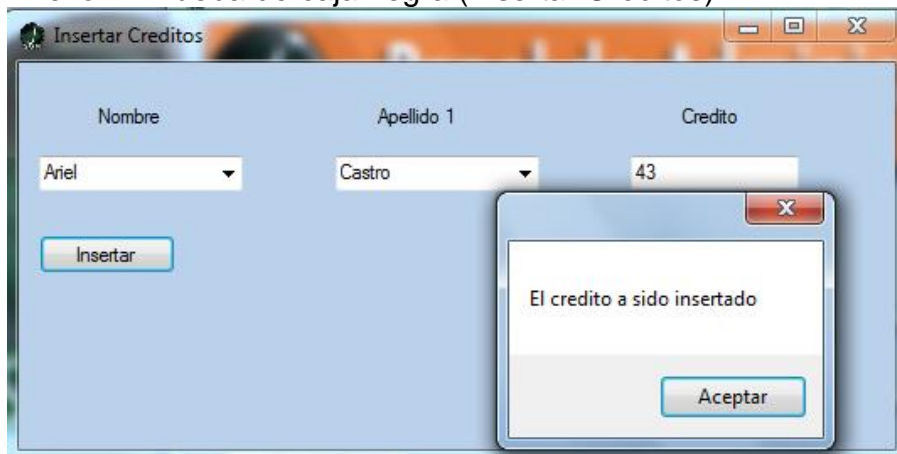


Anexo#3 Vista de los empleados.

# Sistema de Nóminas



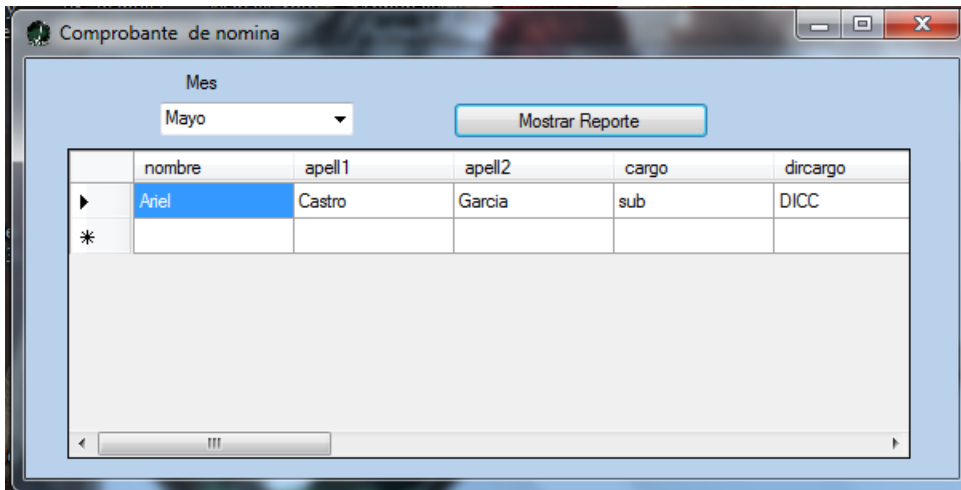
## Anexo#4 Prueba de caja negra (Insertar Créditos).



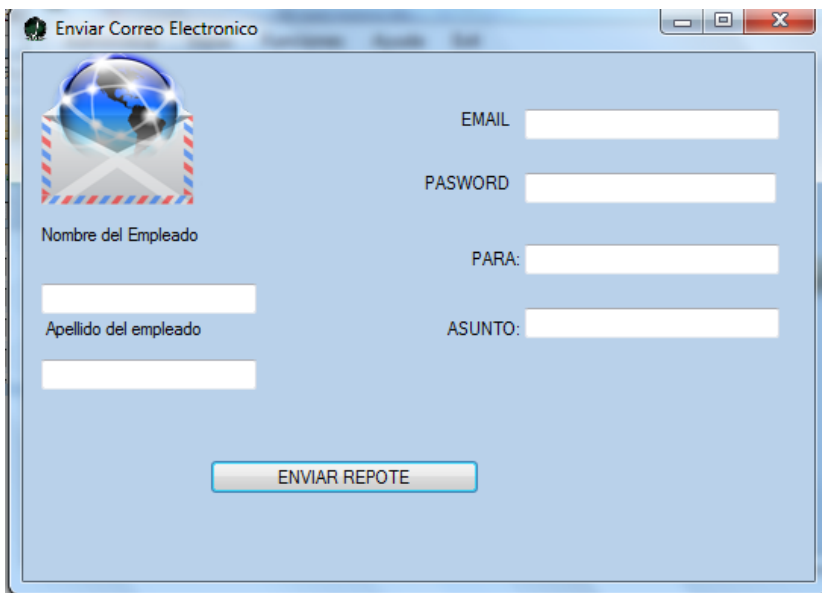
## Anexo #5 Prueba de aceptación (Comprobante de nómina)

# Sistema de Nóminas

---

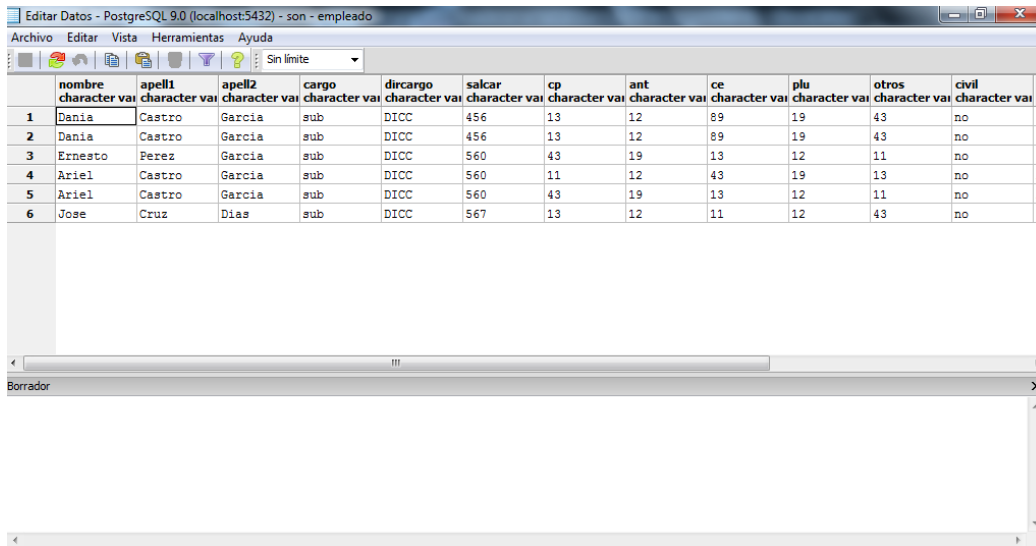


Anexo#6 Funcionalidad Enviar reporte por correo Electrónico.



Anexo# 7 Prueba de consistencia de la Base de Datos.

# Sistema de Nóminas

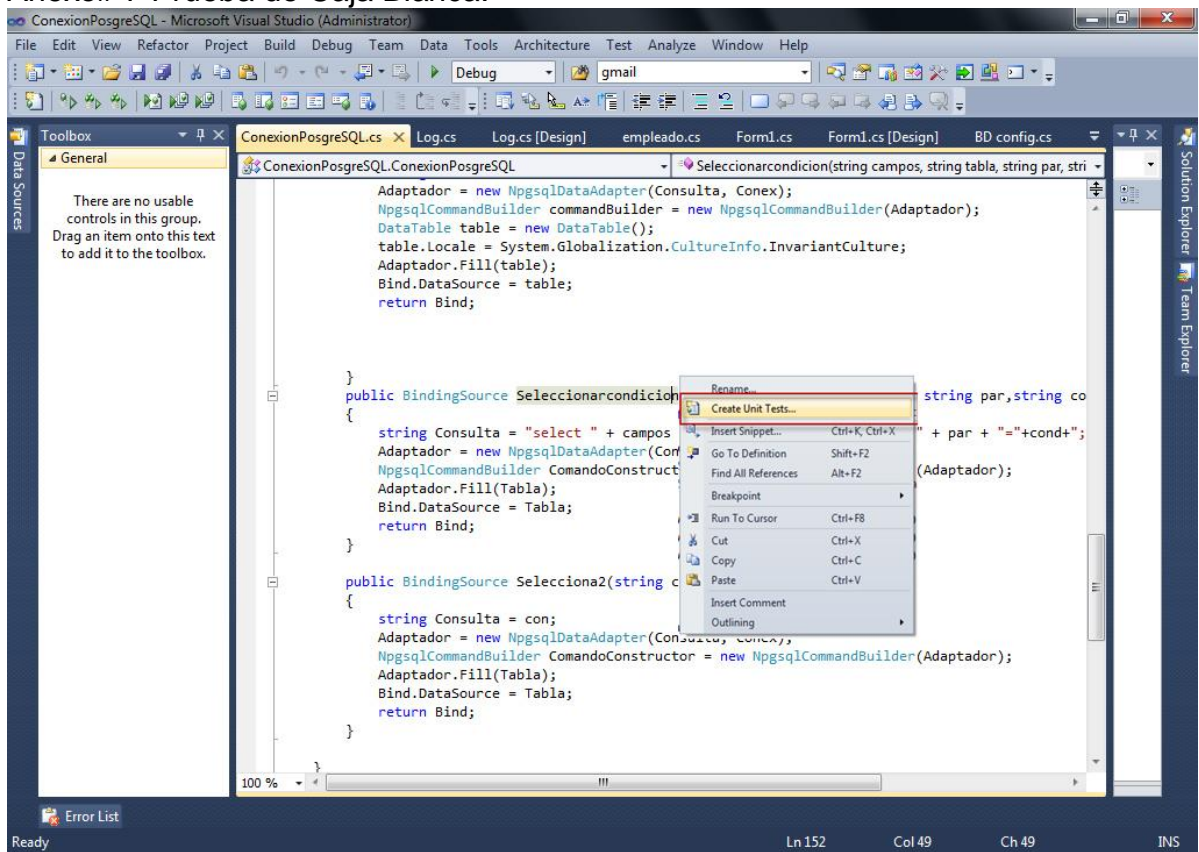


Editar Datos - PostgreSQL 9.0 (localhost:5432) - son - empleado

	nombre character var	apell1 character var	apell2 character var	cargo character var	dircargo character var	salcar character var	cp character var	ant character var	ce character var	plu character var	otros character var	civil character var	d
1	Dania	Castro	Garcia	sub	DICC	456	13	12	89	19	43	no	13
2	Dania	Castro	Garcia	sub	DICC	456	13	12	89	19	43	no	13
3	Ernesto	Perez	Garcia	sub	DICC	560	43	19	13	12	11	no	0
4	Ariel	Castro	Garcia	sub	DICC	560	11	12	43	19	13	no	20
5	Ariel	Castro	Garcia	sub	DICC	560	43	19	13	12	11	no	20
6	Jose	Cruz	Dias	sub	DICC	567	13	12	11	12	43	no	0

Borrador

## Anexo# 7 Prueba de Caja Blanca.



ConexionPosgreSQL - Microsoft Visual Studio (Administrator)

```
File Edit View Refactor Project Build Debug Team Data Tools Architecture Test Analyze Window Help
Debug gmail
Toolbox
Data Sources
General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
ConexionPosgreSQL.ConexionPosgreSQL
Seleccionarcondicion(string campos, string tabla, string par, string con)
Adaptador = new NpgsqlDataAdapter(Consulta, Conex);
NpgsqlCommandBuilder comandoBuilder = new NpgsqlCommandBuilder(Adaptador);
DataTable table = new DataTable();
table.Locale = System.Globalization.CultureInfo.InvariantCulture;
Adaptador.Fill(table);
Bind.DataSource = table;
return Bind;
}
public BindingSource Seleccionarcondicion(string campos, string tabla, string par, string con)
{
    string Consulta = "select " + campos + " from " + tabla + " where " + par + "=" + con + ";";
    Adaptador = new NpgsqlDataAdapter(Consulta, Conex);
    NpgsqlCommandBuilder comandoBuilder = new NpgsqlCommandBuilder(Adaptador);
    Adaptador.Fill(Tabla);
    Bind.DataSource = Tabla;
    return Bind;
}
public BindingSource Selecciona2(string campos, string tabla, string par, string con)
{
    string Consulta = con;
    Adaptador = new NpgsqlDataAdapter(Consulta, Conex);
    NpgsqlCommandBuilder comandoBuilder = new NpgsqlCommandBuilder(Adaptador);
    Adaptador.Fill(Tabla);
    Bind.DataSource = Tabla;
    return Bind;
}
100 %
Error List
Ready Ln 152 Col 49 Ch 49 INS
```