Sistema administrativo para el control de nómina y egresos de la Unidad Educativa Henri Pittier

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Firma

Bárbara Fernández

Caracas, 20/09/2018

Índice General

[Índice de Figuras 3](#_Toc525600422)

[Sinopsis 4](#_Toc525600423)

[Planteamiento del Problema 5](#_Toc525600424)

[Objetivo General 6](#_Toc525600425)

[Objetivos Específicos 6](#_Toc525600426)

[Metodología Empleada 7](#_Toc525600427)

[Desarrollo 9](#_Toc525600429)

[Análisis de los Requerimientos 9](#_Toc525600430)

[Diseño del Sistema 9](#_Toc525600431)

[Implementación de la Base de Datos 9](#_Toc525600432)

[Implementación del Sistema 10](#_Toc525600433)

[Pruebas 10](#_Toc525600434)

[Resultados 11](#_Toc525600435)

[Conclusiones y Recomendaciones 12](#_Toc525600436)

[Referencias Bibliográficas 13](#_Toc525600437)

# Índice de Figuras

[Figura 1 – Fases de la Metodología XP 8](#_Toc525660791)

[Figura 3 – Esquema de patrón Comando 10](#_Toc525660792)

[Figura 4 – Esquema de patrón DAO 11](#_Toc525660793)

[Figura 2 – Modelo Entidad Relación 15](#_Toc525660794)

[Figura 5 - Inicio de sesión de la aplicación 15](#_Toc525660795)

[Figura 6 – Registro de Empleado 16](#_Toc525660796)

[Figura 7 – Consulta de Empleados 16](#_Toc525660797)

[Figura 8 – Modificar Empleados 17](#_Toc525660798)

[Figura 9 – Eliminar Empleados 17](#_Toc525660799)

[Figura 10 - Nómina de Empleados 18](#_Toc525660800)

[Figura 11 – Préstamos de Empleados 18](#_Toc525660801)

# Sinopsis

El presente documento tiene como objetivo expresar el proceso evolutivo del desarrollo del sistema administrativo para el control de nómina y egresos de la Unidad Educativa Henri Pittier, el cual surgió con el fin de automatizar y llevar un registro digital de los pagos y compensaciones de los empleados y a su vez de los datos de cada uno de ellos.

Al ser una institución educativa, cada vez que inicia el nuevo año escolar, se deben obtener los datos de los empleados a trabajar en dicho período académico y establecer el pago de las quincenas correspondientes, de manera que, cada 15 días se deben generar comprobantes de pagos para cada empleado y el respectivo control de egresos de manera manual ya que no se contaba con un sistema que les facilitara el trabajo.

Para solventar esta problemática, se diseñará un sistema administrativo que se adapte a las necesidades de la Unidad Educativa Henri Pittier, almacenando los datos en una base de datos relacional y que sea flexible y amigable para que fomente el uso el mismo.

# Planteamiento del Problema

La unidad educativa Henri Pittier es una institución privada fundada en el 2003, cuenta con aproximadamente 18 empleados y 80 alumnos en edades comprendidas entre 1 y 6 años y se encuentra ubicada en la 3ra avenida de Montalbán I entre calle 1 y 2 en la Quinta San Miguel.

A pesar de tener 15 años de trayectoria, no cuentan con una herramienta capaz de facilitarles el trabajo de cálculos y documentos que realizan de manera manual en todos los períodos académicos.

Esta institución necesita automatizar el control de pago de nómina y de facturación correspondientes a los empleados para poder evaluar el esfuerzo y cantidad de horas trabajadas de cada empleado y de esta manera poder realizar de forma correcta el cálculo para el pago de las compensaciones, y a su vez, llevar un registro digital de los documentos y datos que se manejan para garantizar persistencia en ellos.

## Objetivo General

Desarrollar una aplicación de escritorio para llevar el control de información y gestión de pagos de los representantes y alumnos pertenecientes a la institución U.E Henri Pittier.

## Objetivos Específicos

* Diseñar e implementar una base de datos para almacenar la información del personal docente y administrativo.
* Diseñar e implementar un módulo de inicio de sesión para garantizar que sólo el personal autorizado acceda al sistema.
* Desarrollar un módulo de gestión de información del personal docente y administrativo pertenecientes a la institución.
* Desarrollar un módulo de control de nómina cada empleado con sus respectivos sueldos y salarios.
* Desarrollar un módulo capaz de generar los recibos de pago de los empleados.
* Desarrollar un módulo que genere los comprobantes de egresos.
* Asegurar el acoplamiento de los módulos desarrollados garantizando la unificación del sistema.

# Metodología Empleada

Se hizo uso de la metodología ágil XP o eXtreme Programming, ya que es la que se adapta mejor al trabajo debido a su gran flexibilidad en cuanto a los cambios en los requisitos, ya que los considera algo común o más bien necesario en el desarrollo de un proyecto, por ello, XP es el mejor aliado para desarrollar un sistema cambiante y así obtener un mejor resultado y más aproximado a las necesidades del cliente.

La metodología XP tiene 4 fases, las cuales son:

* 1era Fase o Fase de Planificación del Proyecto: es la primera fase en donde se definen las historias de usuario con el cliente, las mismas ayudan a describir de manera resumida las actividades que puede tener cada actor dentro de la aplicación, se realiza el *Release Planing*, estableciendo las fechas en las que se realizarán las historias de usuario y la prioridad de cada una de ellas, se estiman las iteraciones que por lo general duran aproximadamente 3 semanas, para este proyecto en particular se realizaron iteraciones de 2 semanas (10 días), también se realiza el cálculo de la velocidad del proyecto, que se basa en calcular cuantas historias de usuarios se pueden desarrollar en una iteración y por último se realizan las reuniones diarias en la cual se incluía al cliente para discutir los inconvenientes y determinar posibles soluciones.
* 2da Fase o Fase de Diseño:en esta etapa se realizan los diseños del sistema, se definen los estándares de diseños que serán utilizados tanto en la base de datos, como en la aplicación de escritorio, se establecen los riesgos que pudieran ocurrir y se hace una refactorización del sistema si se cree que se pueden limitar riesgos o funcionalidades extra que luego no serán utilizadas para que, de esta manera, el sistema cumpla de manera adecuada con lo que se pide.
* 3era Fase o Fase de Codificación:luego de haber negociado con el cliente tanto las historias de usuario como los diseños de la aplicación, se procede a desarrollar el sistema. XP opta por la programación en parejas, lo cual es ideal en este caso, ya que la parte restante del sistema fue desarrollada por otro estudiante de la carrera y se tuvo que llegar a un consenso en cuanto a la metodología del sistema, la duración de las iteraciones y los estándares de programación, a su vez, también fomenta el uso de los repositorios para evitar errores de versiones y tener como resultado un código más consistente.
* 4ta Fase o Fase de Pruebas: en esta etapa el objetivo es evaluar todo el sistema y asegurara que se cumplan las historias de usuario que se pactaron con el cliente y a su vez, que los resultados obtenidos sean los esperados, al igual que el resto el código dichas pruebas se trabajan dentro del repositorio para no repetir código.

**Planificación**

**Pruebas**

**Diseño**

**Codificación**

### Figura 1 – Fases de la Metodología XP

# Desarrollo

## Análisis de los Requerimientos

Antes de llevar a cabo todo el desarrollo del proyecto, se analizó cada uno de los requerimientos solicitados por el cliente observando las posibles maneras de realizarlo y si cada uno cumplía en su totalidad las necesidades del negocio. Para ello, se realizó una reunión con el cliente en el cual aceptó los objetivos a desarrollar en el siguiente trabajo.

## Diseño del Sistema

Posteriormente, el primer paso fue diseñar la manera en la que se almacenarían los datos y la interfaz de usuario de la aplicación de escritorio que manejará el cliente, es por ello que se desarrolló el diagrama ER (ver figura 2 en el apéndice) y para el diseño de la interfaz de usuario se tomó como referencia el sistema en Excel que utilizaba antiguamente la institución que contaba con un menú de acceso a diferentes partes del sistema.

## Implementación de la Base de Datos

Para el almacenamiento de datos lo ideal es una base de datos relacional, debido a la reducida cantidad de datos que son necesarios en el negocio. Luego se implementan los *Scripts* correspondientes para implementar dicha base de datos, por último, se realizaron los *Stored Procedures*, o también denominados procedimientos almacenados, que se utilizan para lograr almacenar la información pertinente de manera segura dentro de la base de datos.

## Implementación del Sistema

El desarrollo del proyecto se realizó en Java Version 7 con lenguaje de programación Java y para el diseño de la aplicación se utilizaron herramientas Swing. La estructura del sistema fue de tipo MVC (Modelo, Vista, Controlador) separando las capas del sistema debidamente.

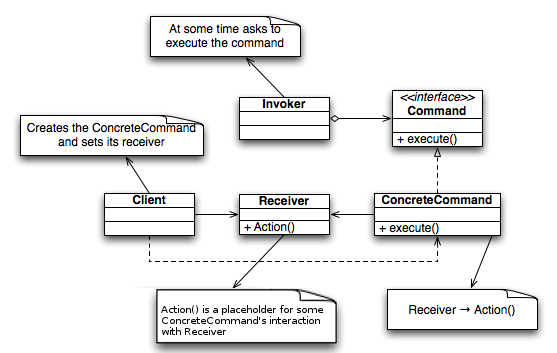
En cuanto al desarrollo de la aplicación de escritorio, existe un inicio de sesión con diferentes permisos (ver figura 5 en el apéndice), ya que es necesario la restricción de algunas zonas, esto se resuelve mediante el uso de dos roles con diferentes permisos, un rol de administrador que pueda acceder al todo el sistema y otro rol secundario, que sólo tenga acceso a algunas regiones específicas del sistema.

En el diseño interno de la aplicación se cuenta con un menú (JMenu) en el cual se puede: registrar, modificar y eliminar un empleado, así como registrar o eliminar un préstamo a alguno de ellos, también se puede generar la nómina de pago, comprobantes de pago y comprobantes de egresos de la institución.

Implementación del Servicio Web

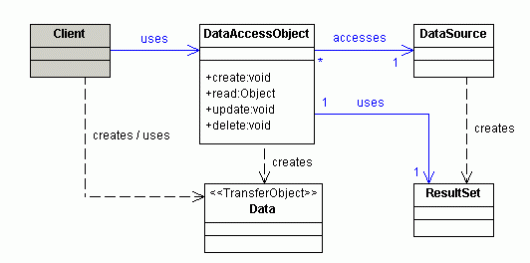
El servicio web es el encargado de recibir las peticiones realizadas por la aplicación de escritorio para analizarlas y realizar las acciones pertinentes para luego dar una respuesta al sistema, en dicho servicio se utilizaron dos tipos de patrones de diseño los cuales fueron:

* Comando: su objetivo es encapsular las acciones posibles que puede realizar el usuario dentro del sistema convirtiéndolas en objetos facilitando la parametrización de los métodos.



### Figura 3 – Esquema de patrón Comando

* DAO: es un conjunto de clases que se encarga de la extracción e ingreso de información a la base de datos, la utilización de este patrón de diseño incrementa la flexibilidad del código debido a que el acceso a los datos sólo se encuentra en las clases de este tipo y al realizar un cambio, las demás clases no sufren modificaciones.



### Figura 4 – Esquema de patrón DAO [Obtenido de https://www.dineshonjava.com/data-access-object/]

## Pruebas

Tanto al finalizar cada etapa de cada iteración como al finalizar el desarrollo del sistema en su totalidad, se realizaron pruebas en el sistema con casos borde y bajo situaciones irregulares probando el funcionamiento del sistema y que el comportamiento fuese de la manera esperada, siempre informando al usuario cuando una situación es fallida o es correcta.

# Resultados

En el presente trabajo se obtuvo como resultado una aplicación amigable capaz de automatizar los procesos de negocio que lleva internamente la Unidad Educativa Henri Pittier, a su vez también se obtuvo una base de datos relacional apta para almacenar la información que es de vital importancia para la institución, que, con el gestionamiento adecuado del servicio web y los procedimientos almacenados (*Stores Procedures*) aseguran la integridad de los datos, lo cual es una tarea de suma importancia ya que en la misma se almacenan datos tanto de pagos de los empleados y su residencia como de los movimientos de la institución.

# Conclusiones y Recomendaciones

Con el presente trabajo se lograron automatizar los pagos de los sueldos y salarios tanto quincenales como mensuales de los empleados, los egresos de la institución y el registro de empleados, así como la modificación y eliminación de los datos de cada uno de ellos, haciendo que el proceso de negocio se realice de manera rápida y limitando el error humano que se percibe cuando dichas tareas se generan de manera manual. Además, se logró almacenar cada uno de los datos que son necesarios de manera digital, evitando gastos en material de oficina y ahorrando el tiempo de búsqueda en los archivos.

Como recomendaciones se debe:

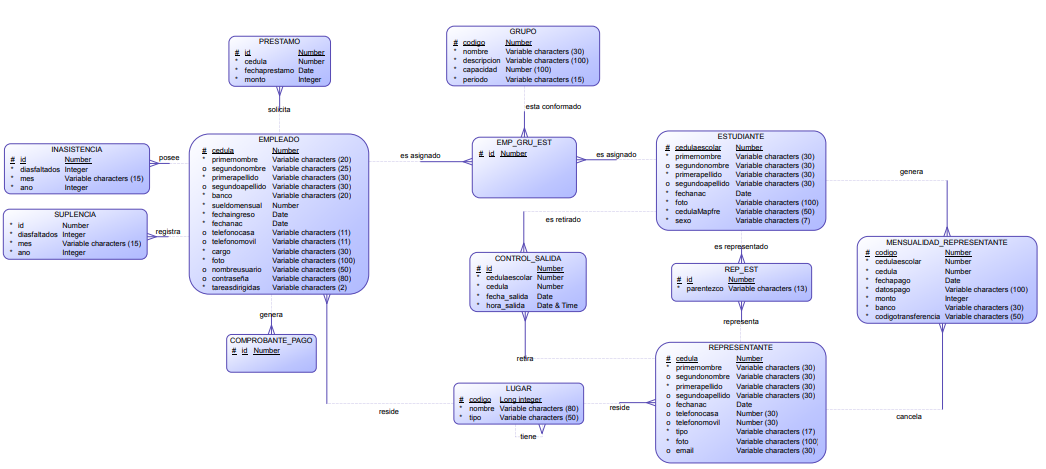
* Realizar mantenimientos al sistema al culminar cada período escolar (cada 10 meses aproximadamente), para asegurar de que el sistema esté funcionando de manera correcta y para eliminar la información innecesaria de la base de datos.
* Al iniciar sesión en los campos de nombre usuario y contraseña colocar un placeholder para que se elimine automáticamente la pista al colocar alguna letra.
* Convertir las imágenes de los empleados y los archivos generados en tipo de dato *BLOB* para posterior almacenamiento en la base de datos.

# Referencias Bibliográficas

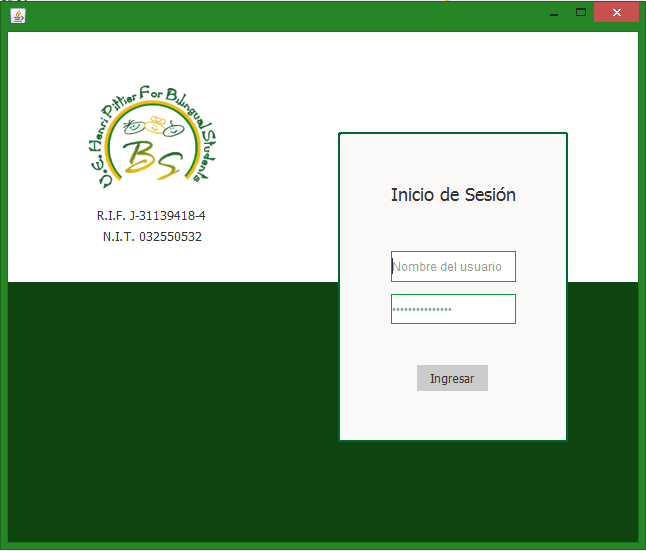
Castillo, O., Daniel, F., & Sevilla, H. (n.d.). *Programacion Extrema.* Retrieved from http://programacionextrema.tripod.com/index.htm

Rajput, D. (2017). *Dinesh Rajput SOftware Engineer.* Retrieved from https://www.dineshonjava.com/data-access-object/

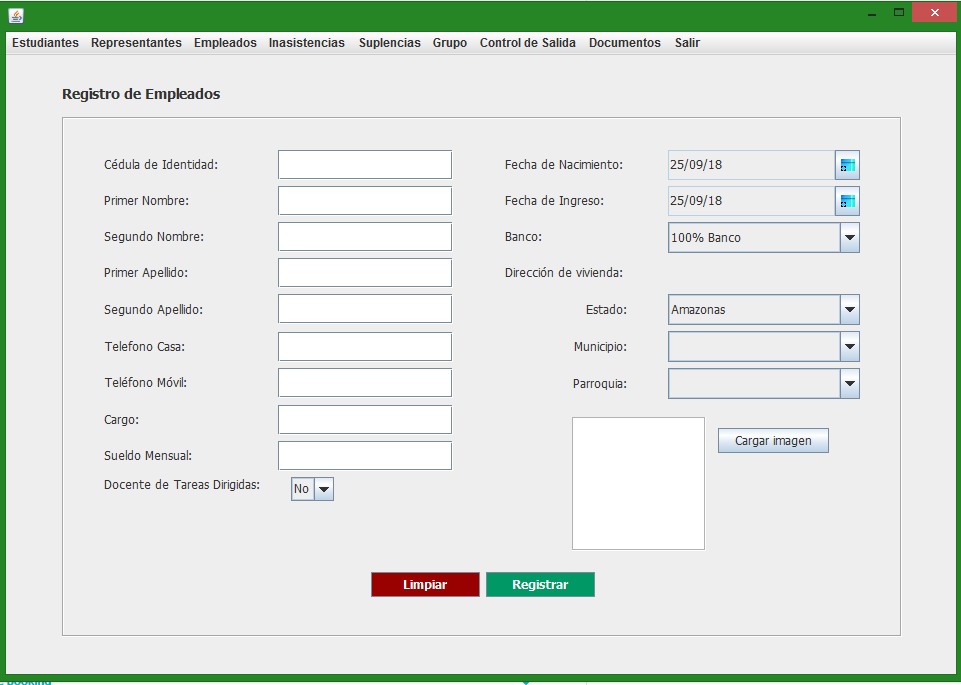
Apéndices



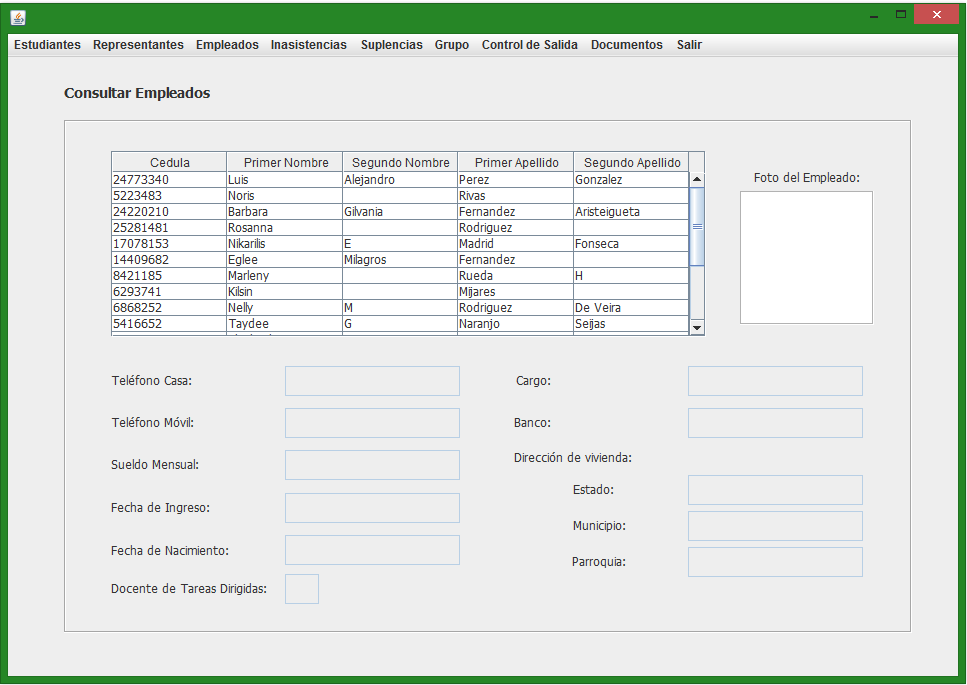
### Figura 2 – Modelo Entidad Relación



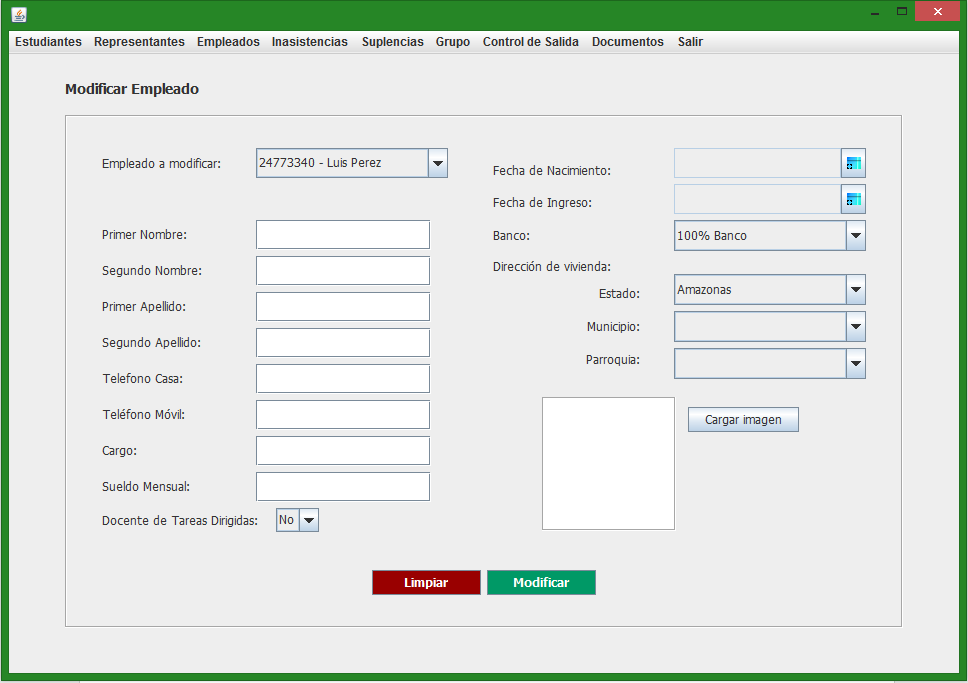
### Figura 5 - Inicio de sesión de la aplicación



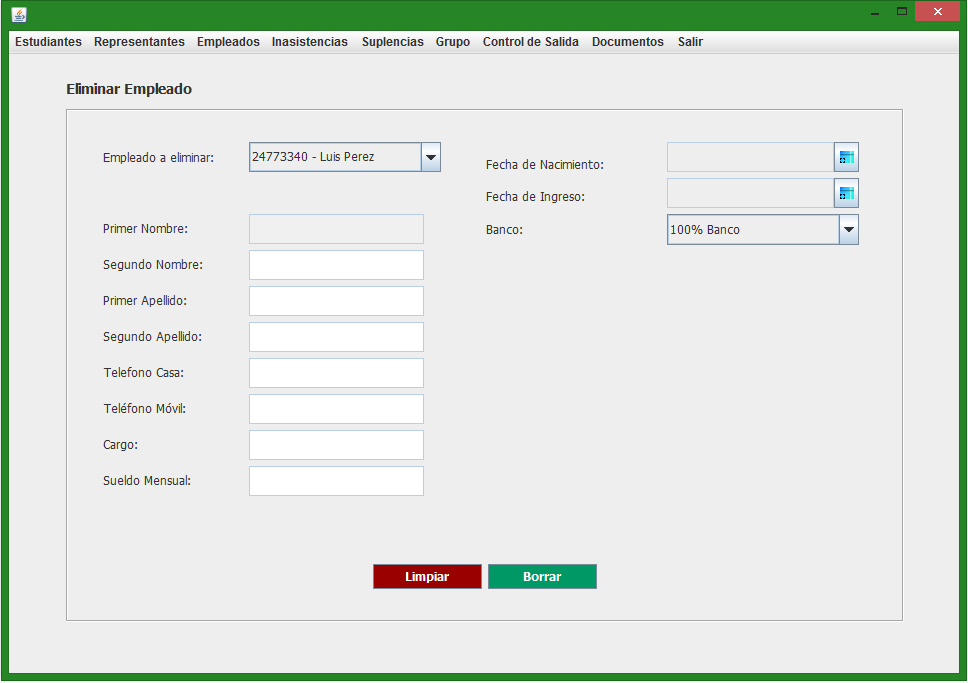
### Figura 6 – Registro de Empleado



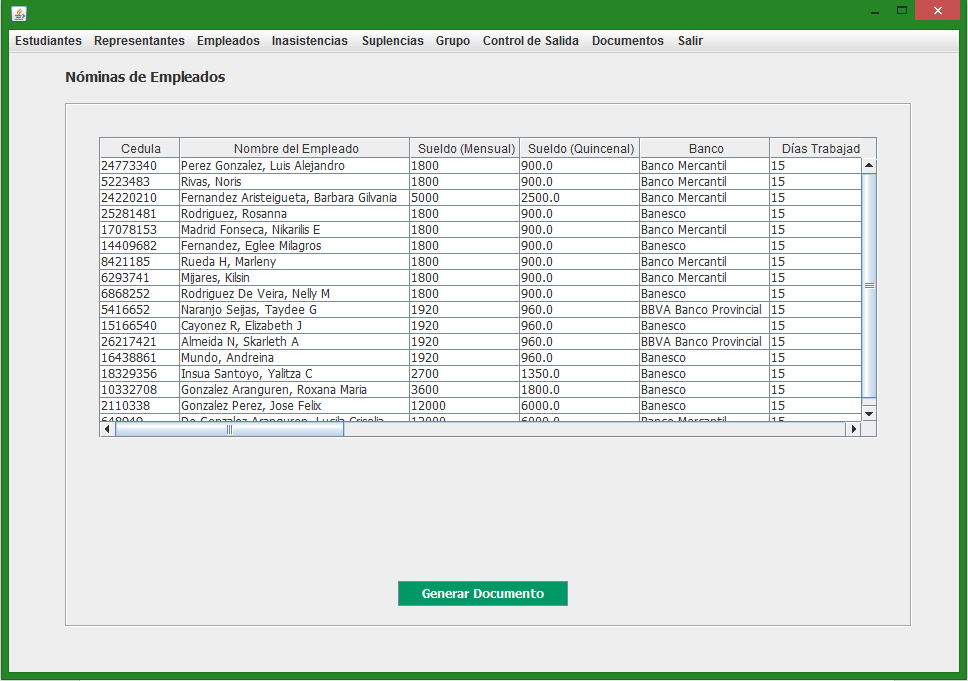
### Figura 7 – Consulta de Empleados



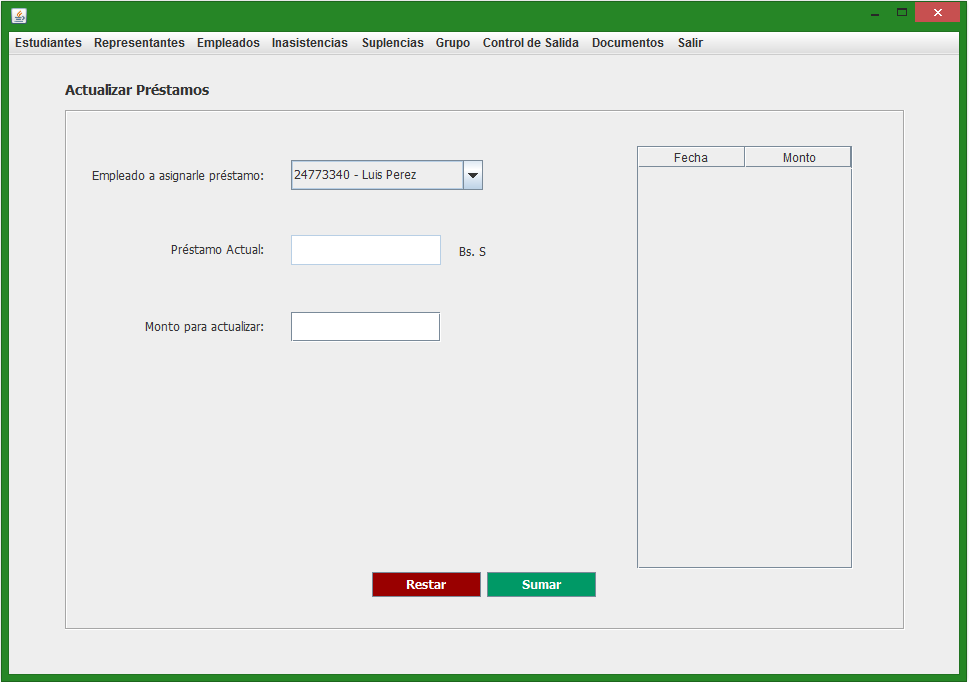
### Figura 8 – Modificar Empleados



### Figura 9 – Eliminar Empleados



### Figura 10 - Nómina de Empleados



### Figura 11 – Préstamos de Empleados