SISTEMA DE NOMINAS

Empresa Consulting S.A

Edwin Chacón – 3090-13-20597

Análisis de sistemas II

Contenido

[Fase de diseño 1](#_Toc54116931)

[Arquitectura 1](#_Toc54116932)

[Patrón de diseño 2](#_Toc54116933)

[Componentes del sistema 3](#_Toc54116934)

[Base de datos 3](#_Toc54116935)

[Modelos 3](#_Toc54116936)

[Controladores 4](#_Toc54116937)

[Interfaces (vistas) 1](#_Toc54116938)

[Reporteria 1](#_Toc54116939)

[Patrón MVC de componentes 2](#_Toc54116940)

[Prototipos del sistema 3](#_Toc54116941)

[Login 3](#_Toc54116942)

[Módulos 3](#_Toc54116943)

[Empleado 4](#_Toc54116944)

[Periodos 4](#_Toc54116945)

[Empresa 5](#_Toc54116946)

[Nominas 5](#_Toc54116947)

[Movimientos 6](#_Toc54116948)

[Métricas para SQA 7](#_Toc54116949)

[Listado de pruebas 7](#_Toc54116950)

[Unitaria 7](#_Toc54116951)

[Integración 8](#_Toc54116952)

[Sistema 8](#_Toc54116953)

[Sistema 8](#_Toc54116954)

[Aceptación 9](#_Toc54116955)

# Fase de diseño

## Arquitectura

Para la operación del sistema se utilizará la arquitectura cliente servidor, a continuación, se explica dicha arquitectura.

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

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un solo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

Una disposición muy común son los sistemas multicapa en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

La red cliente-servidor es una red de comunicaciones en la cual los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc. Este tipo de red puede utilizarse conjuntamente en caso de que se esté utilizando en una red mixta.

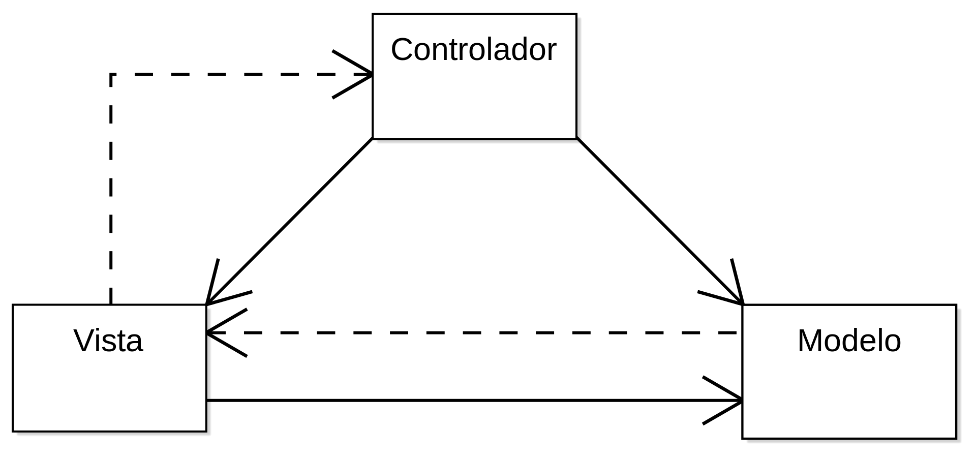


RED

## Patrón de diseño

Se utilizará la modelo vista controlador para la creación del sistema, esto para realizar una aplicación escalable y con separación de responsabilidades, a continuación se detalla el modelo.

Modelo-vista-controlador (MVC) es un patrón de arquitectura de software, que separa los datos y principalmente lo que es la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. ​



## Componentes del sistema

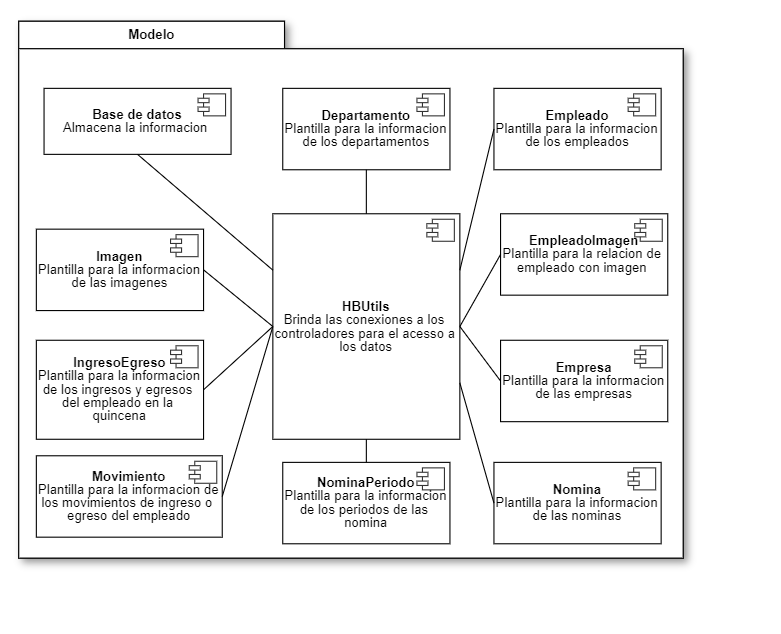
### Base de datos

Encargada de almacenar y brindar seguridad a la información del sistema.

* HBUtils

### Modelos

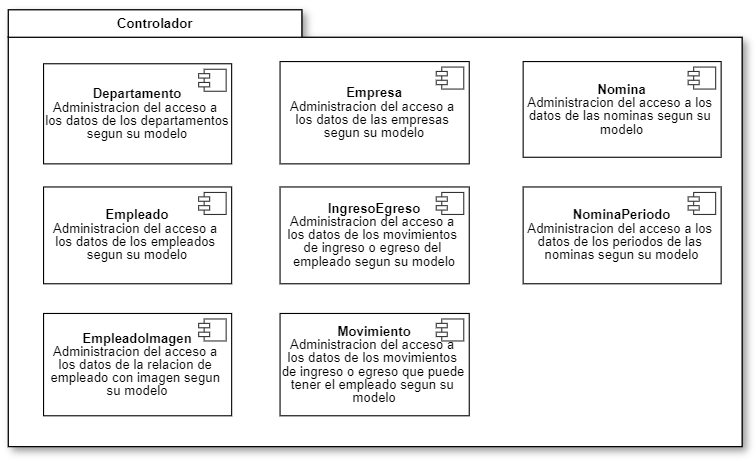
* Departamento
* Empleado
* EmpleadoImagen
* Empresa
* Imagen
* IngresoEgreso
* Movimiento
* Nomina
* Periodo



### Controladores

Serán los encargados de comunicarse y gestionar el acceso a la base de datos

* Departamento
* Empleado
* EmpleadoImagen
* Empresa
* IngresoEgreso
* Movimiento
* Nomina
* Periodo

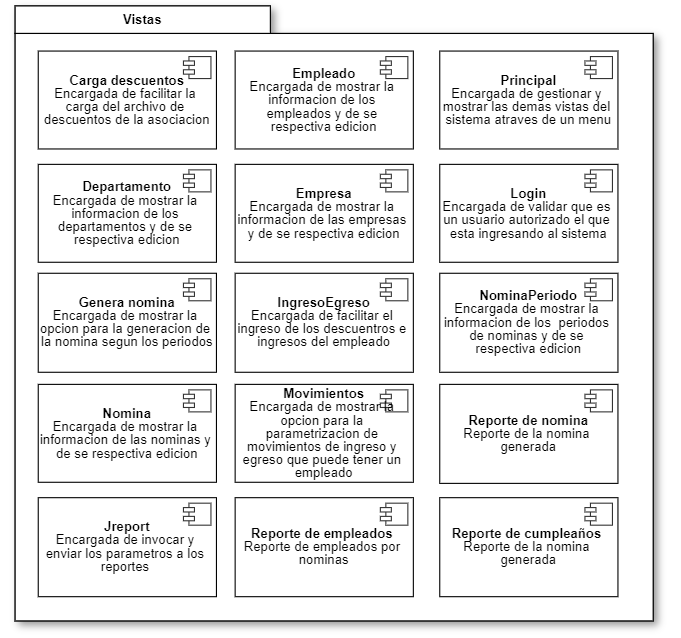


### Interfaces (vistas)

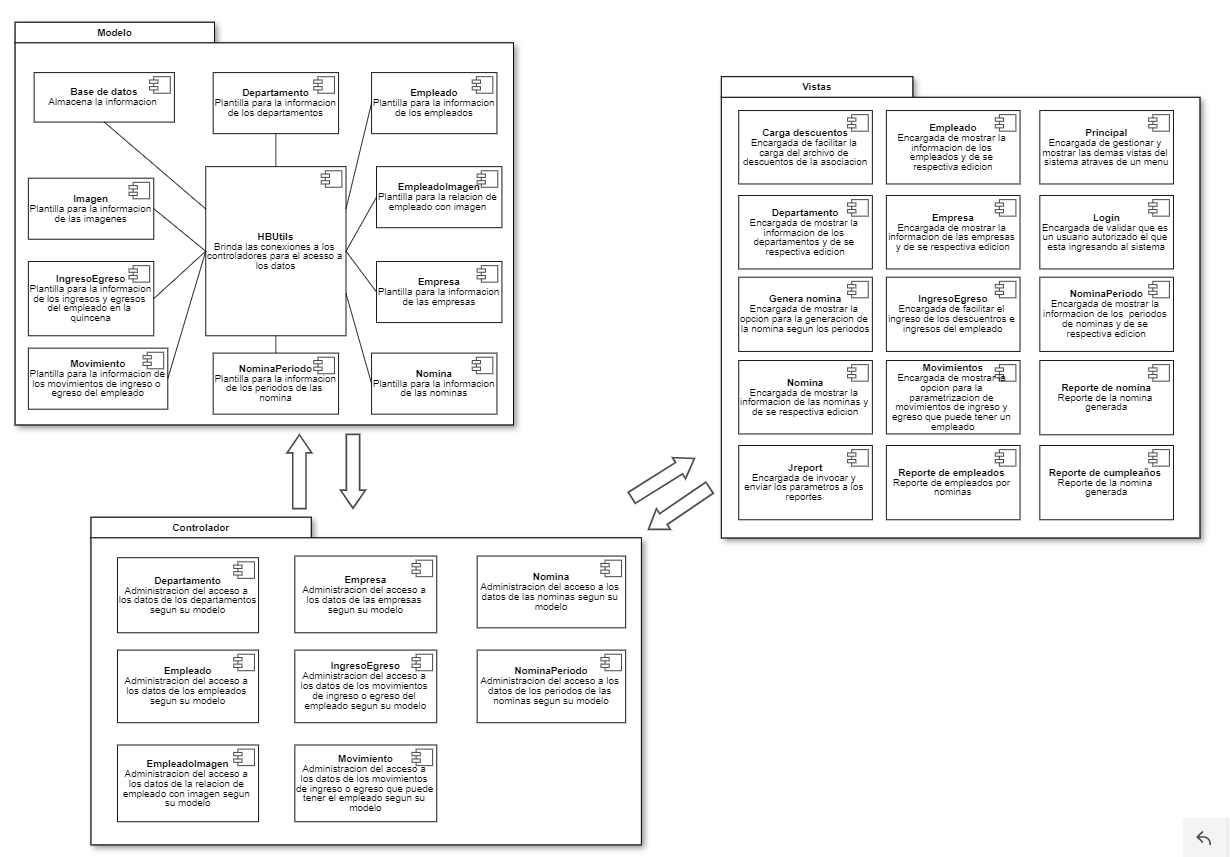
* Carga descuentos
* Departamento
* Empleado
* Empresa
* Genera nomina
* IngresoEgreso
* Login
* Movimiento s
* Nomina
* Periodo
* Principal

### Reporteria

* JRerport
* ReportEmpleado
* ReportNomina
* ReportCumpleanios



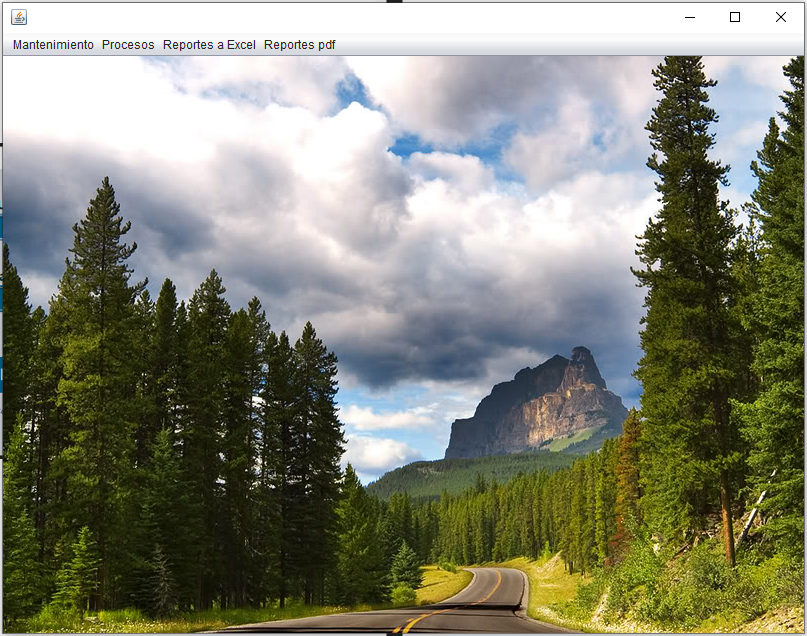
### Patrón MVC de componentes



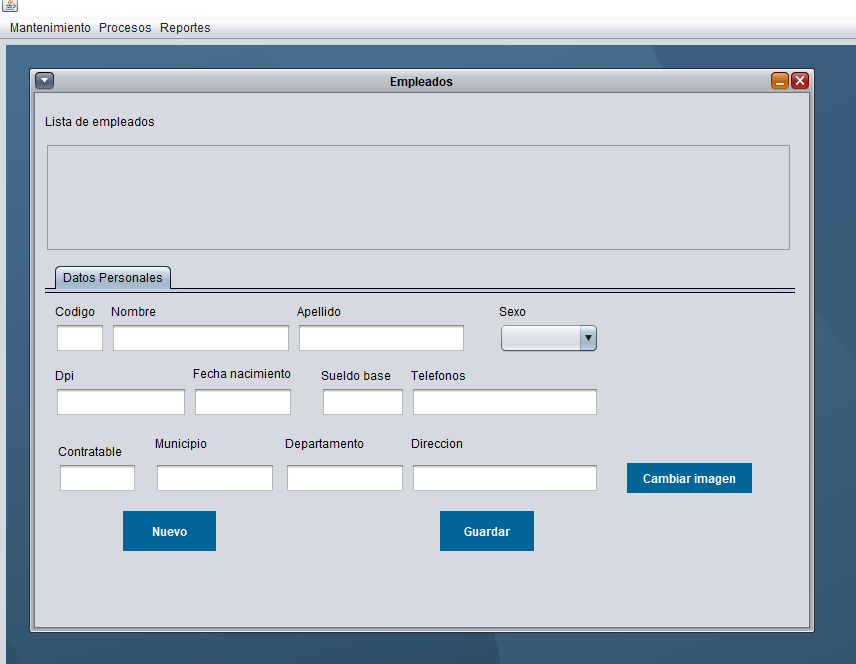
## Prototipos del sistema

### Login

### Módulos

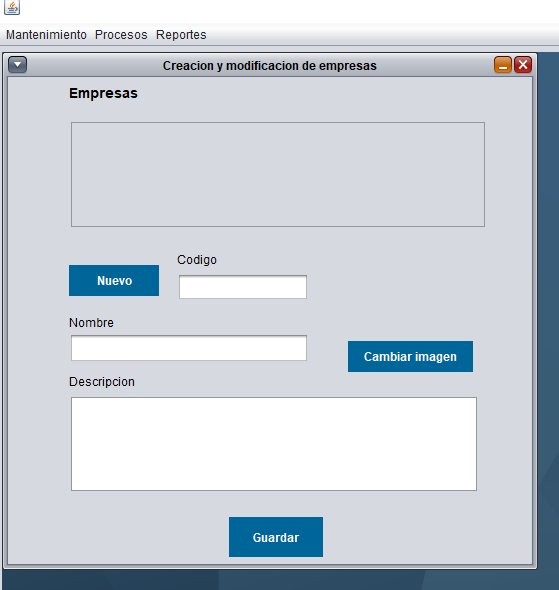


### Empleado



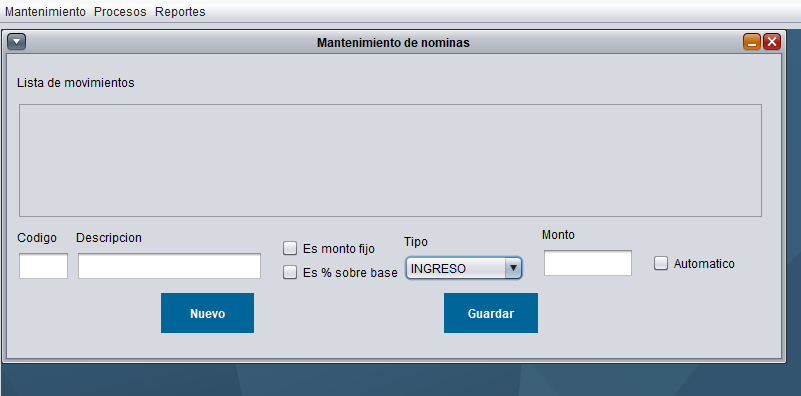
### Periodos

### Empresa



### Nominas

### Movimientos



## Métricas para SQA

### Listado de pruebas

|  |  |
| --- | --- |
| Prueba | Descripción |
| Unitaria | Funcionalidad a los componentes |
| Integración | Comunicación y sincronización entre los distintos componentes del sistema |
| Sistema | Correcto funcionamiento de los componentes |
| Implantación | Funcionamiento y puesta en marcha den producción |
| Aceptación | Cumplimiento de los requerimientos solicitados |

### Unitaria

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clave | SQA | | Prueba | Descripción | |
| P1 | Oscar García | | Unitaria | Funcionalidad a los componentes | |
| Acción | | **Resultado Esperado** | | | **Resultado Obtenido** |
| Que los tipos de datos sean los correctos | | Tipos de datos según la base de datos | | |  |
| Funcionamiento de las funciones de acceso a la base de datos | | Que las funciones cumplan su objetivo | | |  |
| Funcionamiento de las funciones auxiliares de los componentes | | Que las funciones cumplan su objetivo | | |  |
| Optimización de código | | Que las funciones sean reutilizables | | |  |
| Hardcode | | Que no hayan datos harcodeados sino lo amerita el sistema | | |  |

### Integración

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clave | SQA | | Prueba | Descripción | |
| P2 | Oscar García | | Integración | Comunicación y sincronización entre los distintos componentes del sistema | |
| Acción | | **Resultado Esperado** | | | **Resultado Obtenido** |
| Los catálogos se integran (empresa, periodos, departamentos, nomina, movimientos) | | Los catálogos se integran en los distintos componentes donde son utilizados y se obtienen de la base de datos | | |  |

### Sistema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clave | SQA | | Prueba | Descripción | |
| P3 | Oscar García | | Sistema | Correcto funcionamiento de los componentes | |
| Acción | | **Resultado Esperado** | | | **Resultado Obtenido** |
| Los componentes cumplen su función | | Los componentes cumplen la función para la cual fueron creados | | |  |
| Los componentes procesan sus errores | | Los componentes tienen la capacidad de procesar los errores sin hacer que la aplicación falle | | |  |

### Sistema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clave | SQA | | Prueba | Descripción | |
| P3 | Oscar García | | Implantación | Funcionamiento y puesta en marcha den producción | |
| Acción | | **Resultado Esperado** | | | **Resultado Obtenido** |
| instalación del sistema | | El sistema se puede instalar en las maquinas donde va a ser utilizado | | |  |
| Producción | | El sistema funciona correctamente en un ambiente de producción | | |  |
| Funcionamiento fuera del pc del programador | | El sistema funciona correctamente en pc que no sea del programador. | | |  |

### Aceptación

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clave | SQA | | Prueba | Descripción | |
| P4 | Oscar García | | Aceptación | Cumplimiento de los requerimientos solicitados | |
| Acción | | **Resultado Esperado** | | | **Resultado Obtenido** |
| Requerimientos | | El sistema cumple con el análisis de requerimientos aceptado por el cliente. | | |  |