**UNIVERSIDAD AUTÓNOMA GABRIEL RENE MORENO**

*Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones*



**Primer parcial Tecnología Web**

**“SISTEMAS de Marketing y Ventas (INEGAS)”**

**Estudiantes:**

Montero Figueroa Juan Fernando 212080202

Morochi Sanca Isaias 213012741

Yucra Serrudo Juan Pablo 214059588

**Materia/Sigla:** Tecnología Web / INF513

**Grupo:**  SA

**Docente:**  Ing. Evans Balcazar Veizaga

Santa Cruz – Bolivia

Contenido

[Introducción 5](#_Toc529183072)

[Capítulo 1: Perfil del proyecto 6](#_Toc529183073)

[1.1 Definición del problema 6](#_Toc529183074)

[1.1.1 Situación problemática 6](#_Toc529183075)

[1.1.2 Situación deseada 6](#_Toc529183076)

[1.2 Objeto de investigación 7](#_Toc529183077)

[1.3 Objetivos 7](#_Toc529183078)

[1.3.1 Objetivo general 7](#_Toc529183079)

[1.3.2 Objetivos específicos 7](#_Toc529183080)

[1.4 Metodología 8](#_Toc529183081)

[1.5 Alcance 9](#_Toc529183082)

[1.5.1 Requisitos funcionales. 9](#_Toc529183083)

[1.5.2 Requisitos no funcionales. 10](#_Toc529183084)

[Capítulo 2: Captura de requisitos 11](#_Toc529183085)

[3.1 Identificar y describir actores y casos de uso 11](#_Toc529183086)

[3.1.1 Actores 11](#_Toc529183087)

[3.1.2 Casos de uso 11](#_Toc529183088)

[3.2 Priorización de casos de uso 11](#_Toc529183089)

[3.3 Especificación de casos de uso 12](#_Toc529183090)

[CU1. Gestión de usuario 12](#_Toc529183091)

[CU2. Gestión de programas 13](#_Toc529183092)

[CU3. Gestión de oferta 14](#_Toc529183093)

[CU4. Gestión de reservas 15](#_Toc529183094)

[CU5. Gestión de ventas 16](#_Toc529183095)

[CU6. Gestión de mensajes y notificaciones 17](#_Toc529183096)

[CU7. Gestión de redes sociales 18](#_Toc529183097)

[CU8. Reportes y estadística 19](#_Toc529183098)

[3.4 Modelo general de casos de uso 20](#_Toc529183099)

[Capítulo 4: Análisis 21](#_Toc529183100)

[4.1 Análisis de la arquitectura 21](#_Toc529183101)

[4.1.1 Identificación de paquetes 21](#_Toc529183102)

[4.1.2 Relación de paquetes y casos de uso 21](#_Toc529183103)

[4.2 Análisis de casos de uso 23](#_Toc529183104)

[4.3 Análisis de paquetes 26](#_Toc529183105)

[Capítulo 5: Diseño 27](#_Toc529183106)

[5.1 Diseño de la arquitectura 27](#_Toc529183107)

[5.1.1 Arquitectura lógica 27](#_Toc529183108)

[5.1.2 Arquitectura física 28](#_Toc529183109)

[5.2 Diseño de la base de datos 29](#_Toc529183110)

[5.2.1 Diseño conceptual 29](#_Toc529183111)

[5.2.2 Diseño lógico 30](#_Toc529183112)

[5.2.3 Diseño físico 32](#_Toc529183113)

[Script 36](#_Toc529183114)

[5.4 Diseño de casos de uso 40](#_Toc529183115)

[5.3.1 Diagramas de secuencia 40](#_Toc529183116)

[5.3.2 Diagramas de detalle procedimental 47](#_Toc529183117)

[Capítulo 6: Implementación 50](#_Toc529183118)

[6.1 Plataforma de desarrollo de software 50](#_Toc529183119)

[6.1.1 Lenguaje de programación 50](#_Toc529183120)

[6.1.2 Base de datos 50](#_Toc529183121)

[6.1.3 Sistemas operativos 51](#_Toc529183122)

[6.1.4 Otros 51](#_Toc529183123)

[Capítulo 7: Pruebas 53](#_Toc529183124)

[CU1. Gestionar usuarios 53](#_Toc529183125)

[CU2. Gestionar evento 53](#_Toc529183126)

[CU3. Gestion de Oferta 54](#_Toc529183127)

[CU4. Gestion de Reserva 54](#_Toc529183128)

[CU5. Gestion Venta 55](#_Toc529183129)

[CU6. Gestion de Notificaciones 55](#_Toc529183130)

[CU7. Gestion de Redes Sociales 55](#_Toc529183131)

[CU8. Reportes y estadística 56](#_Toc529183132)

[Bibliografía 56](#_Toc529183133)

# Introducción

Los cursos que ofrecen Actualmente el número de eventos que realiza la población, entre ellos cumpleaños, aniversarios, fiestas de promociones, fiestas de graduaciones, entre otros, va en aumento y junto a ello igual aumenta la demanda que se tiene a los estudios fotográficos para que estos puedan asistir a dichos eventos y tomar las respectivas fotografías tanto para los invitados como para los anfitriones. Esto provoca que los estudios fotográficos tengan una mayor cantidad de trabajos, lo cual puede generar problemas de organización si es que el estudio no cuenta con una buena administración. Es por esta razón que se considera necesario el desarrollo de un software que permita administrar de forma sencilla los procedimientos realizados dentro de un estudio fotográfico.

# Capítulo 1: Perfil del proyecto

## 1.1 Definición del problema

El área de marketing y ventas (Inegas) por lo general manejan una gran cantidad de información, acerca de eventos que ofrece como ser: cursos, conferencias y talleres. Esto genera gran flujo de información acerca de los cursos acerca de coordinación con los días, horarios coordinación y administración razón por la cual se considera necesario el desarrollo de un software para que el área de marketing y ventas pueda administrar sus procedimientos de forma sencilla a su vez que se le ofrece a los clientes un medio por el cual pueda solicitar los información de los cursos, talleres o conferencias y poder reservar su asistencia sin la necesidad de presentarse físicamente al establecimiento.

### 1.1.1 Situación problemática

El área de Marketing y Ventas cuenta con problemas de administración al momento de proporcionar información y registro de clientes debido a la falta de organización, ocasionando problemas en el desarrollo de sus actividades y generando pérdidas.

### 1.1.2 Situación deseada

Tener un manejo más eficiente al momento de proporcionar información de sus servicios que por medio de un software sean capaces de administrar la parte de ofertas, ventas, reservas y otros. Así de esta manera realizar de forma sencilla, permitiendo un mejor uso de su personal y recursos.

## 1.2 Objeto de investigación



## 1.3 Objetivos

### 1.3.1 Objetivo general

* Desarrollar un software para la administración de sus actividades, recepción de solicitudes de información y reserva para un taller, conferencia o cursos por medio de un servicio de correo electrónico.

### 1.3.2 Objetivos específicos

* Definir las funcionalidades que deberá tener el software para satisfacer las necesidades de los clientes.
* Diseñar la arquitectura que tendrá el software y el flujo de proceso que se seguirá para cumplir cada uno de los requerimientos establecidos en el análisis.
* Implementar cada uno de los requerimientos siguiendo modelos y diseños establecidos.
* Realizar las pruebas correspondientes para verificar el correcto funcionamiento del software o detectar posibles errores.

## 1.4 Metodología

Se utilizará como estrategia y como métodos al Proceso Unificado de Desarrollo de Software (PUDS) específicamente de (Booch, Rumbaugh, &Jacobson, 2000) para el desarrollo del sistema, ya que es una metodología centrada en la arquitectura, y dirigida por casos de uso, como fue definida por sus autores Boch, Jacobson y Rumbaugh, y por esas razones es adecuada para el desarrollo de este sistema de gestión.

Además, se utilizará al Lenguaje Unificado de Modelado (UML) como lenguaje de representación de los modelos resultantes en cada actividad del PUDS. A continuación, veremos las "Fases", "Flujos de Trabajo", y "Actividades" correspondientes al PUDS que se llevarán a cabo durante el desarrollo del sistema.

* **Fase de Inicio:**
  + **Captura de Requisitos**
* Identificar y describir actores y casos de uso.
* Priorización de casos de uso.
* Detallar casos de uso.
* Realizar los prototipos de interface de los casos de uso identificados.

**Resultado:** Modelo de dominio, Modelo de casos de uso.

* **Fase de elaboración:**
  + **Análisis**
* Análisis de la arquitectura.
* Análisis de casos de uso.
* Análisis de paquetes.

**Resultado:** Especificación de requisitos de software.

* + **Diseño**
  + Diseño de la arquitectura.
  + Diseño de casos de uso
  + Diseño de datos.

**Resultado:** Descripción de diseño de software.

* **Fase de Construcción:**
  + **Implementación**
* Traducción de los modelos de diseño a lenguaje de programación.

**Resultado:** Modelo de implementación, Código fuente.

## 1.5 Alcance

### 1.5.1 Requisitos funcionales.

**RF1: Gestión de usuario (administrador, cliente)**

Este requisito permitirá que se pueda registrar, modificar y eliminar usuarios en el software.

**RF2: Gestión de evento**

Este requisito permitirá que un cliente pueda hacer consulta de los eventos, costos, etc. que se realizaran que solicite el cliente.

**RF3: Gestión de ofertas**

Este requisito permitirá que se puedan tener conocimiento de los distintos eventos que se presentaran tanto como ser duración, fecha de inicio y horarios para que el cliente pueda apreciar y escoger el más oportuno que le parezca.

**RF4: Gestión de reservas**

Este requisito permitirá que se puedan registrar, permitiéndole proporcionarle información sobre el evento que le intereso o promociones de otros eventos de esta manera para poder mostrar los servicios que ofrece.

**RF5: Gestión de ventas**

Este requisito permitirá que puedan confirmar su inscripción al curso cancelando el costo de curso y enviando el número de comprobante del depósito evitando ir hasta su predios, lo cual el cliente se resultara más fácil ofreciéndoles un mejor atención al cliente al momento de realizar su inscripción lo cual podrán ser usados por los clientes cuando se registren un evento.

**RF6: Gestión de mensajes y notificaciones**

Este requisito permitirá enviarles un recordatorio tantos a los administradores como a los distintos clientes que se registraron a un evento.

**RF7: Gestión de Redes Sociales**

Este requisito permitirá que los clientes puedan consultar sobre el costo total que tendría registrar un nuevo evento o pedido junto con sus respectivos servicios.

**RF8: Reportes y estadística**

Este requisito permitirá que se puedan obtener gráficos y tablas estadísticas las cuales mostraran información referente a los eventos realizados respectos a los cursos, talleres y conferencias.

### 1.5.2 Requisitos no funcionales.

#### 1.5.2.1 Seguridad.

Funcionalidad encargada de mantener coherente la base de datos de registro del perfil profesional. En caso de eventualidades no deseadas, realizando copias de seguridad y restauración. Los módulos que participan son:

* Validación de entrada de datos
* Validación de accesos
* Restauración de base de datos
* Backus de la base de datos

# Capítulo 2: Captura de requisitos

## 3.1 Identificar y describir actores y casos de uso

Este capítulo tiene como propósito la captura de requisitos, describir en forma textual los requisitos, identificar actores, identificar casos de uso, detallar casos de uso y definir la estructura del modelo general de casos de uso.

### 3.1.1 Actores

* **Cliente:** Son los usuarios que harán uso del software para poder registrar eventos y solicitar información acerca de los cursos, talleres y conferencias.
* **Administrador:** Son los usuarios que se encargaran de atender los clientes que lleguen al software.

### 3.1.2 Casos de uso

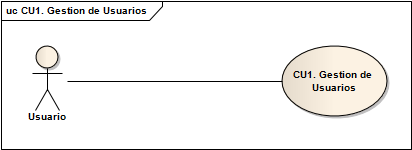
* CU1. Gestión de Usuario
* CU2. Gestión de programa
* CU3. Gestión de oferta
* CU4. Gestión de reservas
* CU5. Gestión de ventas
* CU6. Gestión de mensajes y notificaciones
* CU7. Gestión de redes sociales
* CU8. Reportes y estadística

## 3.2 Priorización de casos de uso

|  |  |
| --- | --- |
| **Caso de uso** | **Priorización** |
| CU1. Gestión de usuario | Alto |
| CU2. Gestión de programa | Alto |
| CU3. Gestión de oferta | Medio |
| CU4. Gestión de reservas | Medio |
| CU5. Gestión de ventas | Alto |
| CU6. Gestión de mensajes y notificaciones | Alto |
| CU7. Gestión de redes sociales | Bajo |
| CU8. Reportes y estadística | Alto |

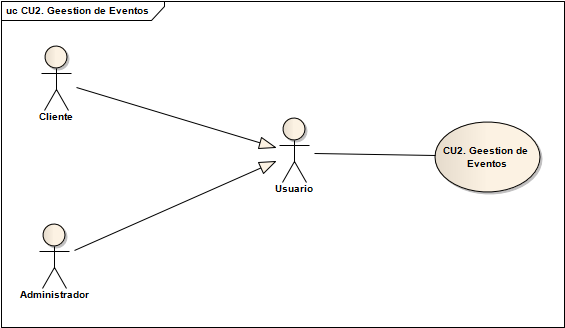
## 3.3 Especificación de casos de uso

### CU1. Gestión de usuario



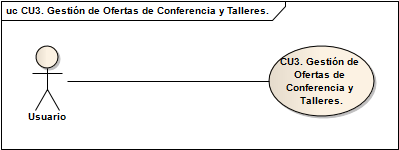
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de usuario |
| Propósito | Esta funcionalidad permitirá que los usuarios que vayan a hacer uso del software puedan registrarse. |
| Actores | Cliente, Administrador |
| Actor iniciador | Cliente, Administrador |
| Pre condición | Ninguna |
| Flujo principal | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo se podrá registrar un usuario y especificar si el usuario será cliente interesado o cliente confirmado.   MODIFICAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectico el administrador podrá hacer modificaciones sobre la información del cliente.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo el administrador podrá eliminar un usuario del sistema, especificando el ID del usuario. |

### CU2. Gestión de programas



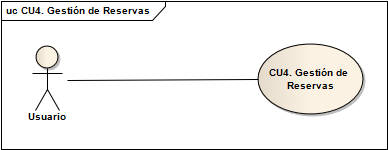
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de programas |
| Propósito | Esta funcionalidad permitirá crear a los administradores puedan registrar un nuevo evento o modificar y eliminar un evento que hubieran registrado anteriormente. De igual manera, permitirá poder obtener un listado con todos los eventos registrado en el software |
| Actores | Administrador |
| Actor iniciador | Administrador |
| Pre condición | Ninguna |
| Flujo principal | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder registrar un evento.   MODIFICAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder modificar un evento.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder modificar un evento, especificando el ID del evento.   AGREGAR UN EVENTO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder agregar un servicio a un determinado evento, especificando el id del servicio y el id del evento.   QUITAR UN EVENTO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder quitar un determinado evento, especifican el id del evento. |
| Post condición | Ninguna |
| Excepción | La fecha indicada para el evento deberá ser una fecha mayor o igual a la fecha del día en que se haya realizado el registro. |

### CU3. Gestión de oferta



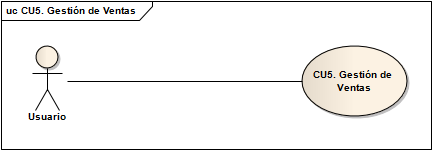
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de ofertas |
| Propósito | Esta funcionalidad del software que permitirá conocer las ofertas de conferencias y talleres mostrando el cronograma de los cursos horarios, fecha y tema. |
| Actores | Cliente, Administrador |
| Actor iniciador | Cliente, Administrador |
| Pre condición | Ninguna |
| Flujo principal | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se podrá registrarán los distintos eventos que se ofertarán.   MODIFICAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se podrá cambiar las fechas, horarios, lugar de dichos eventos.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se podrá eliminará los eventos pasados. |
| Post condición | Ninguna |
| Excepción | Ninguna |

### CU4. Gestión de reservas



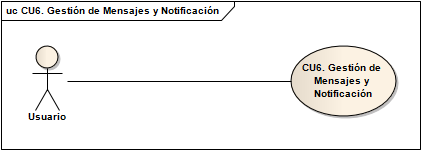
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de reservas |
| Propósito | Esta funcionalidad permitirá al cliente que puedan reservar un cupo para el curso, taller o conferencia que le interese. |
| Actores | Cliente |
| Actor iniciador | Cliente |
| Pre condición | Ninguna |
| Flujo principal | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder realizar el registro del curso, taller o conferencia que le interese al cliente. |
| Post condición | Ninguna |
| Excepción | Ninguna |

### CU5. Gestión de ventas



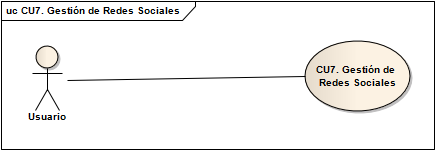
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de ventas |
| Propósito | Esta funcionalidad permitirá que se puedan registrar permanentemente una vez hecho el pago en sus entidades autorizadas. |
| Actores | Cliente |
| Actor iniciador | Cliente |
| Pre condición | Ninguna |
| Flujo principal | AGREGAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder registrar un curso, taller o conferencia enviando el número de comprobante cancelado. |
| Post condición | Ninguna |
| Excepción | No se registrara comprobantes duplicados. |

### CU6. Gestión de mensajes y notificaciones



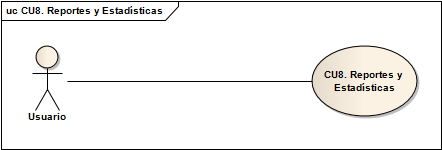
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de mensajes y notificaciones |
| Propósito | Esta funcionalidad permitirá que el software pueda enviar un recordatorio como mensajes a los administradores y notificaciones a los clientes acerca del curso. |
| Actores | Administrador |
| Actor iniciador | Administrador |
| Pre condición | Que este registrado |
| Flujo principal | AGREGAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se agregará una notificación con un asunto, descripción y tipo.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se eliminará notificaciones de las cuales ya no se utilizarán o de eventos anteriores.   ENVIAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo se enviar la notificación del tiempo restante que falta para que el curso, taller o conferencia inicie y puedan asistir a tiempo. |
| Post condición | Ninguna |
| Excepción | Ninguna |

### CU7. Gestión de redes sociales



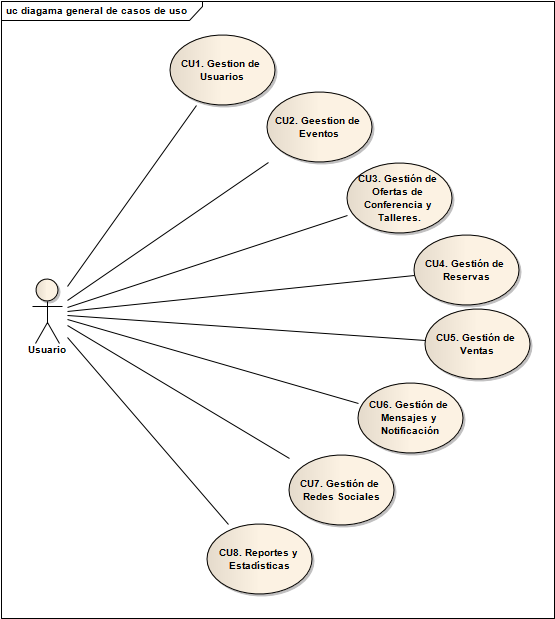
|  |  |
| --- | --- |
| Nombre de caso de uso | Gestión de redes sociales |
| Propósito | Esta funcionalidad permitirá que los cliente puedan realizar proformas para poder conocer el costo que tendría registrar un evento o un pedido con sus respectivos servicios. |
| Actores | Cliente |
| Actor iniciador | Cliente |
| Pre condición | Ninguna |
| Flujo principal | REGISTRAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo para poder realizar el registro de una nueva proforma.   ELIMINAR   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo para poder eliminar una proforma registrada en el sistema.   AGREGAR UN SERVICIO   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto el comando respectivo para poder agregar un nuevo servicio a la proforma, especificando el ID de la proforma y el ID del servicio. |
| Post condición | Ninguna |
| Excepción | Ninguna |

### CU8. Reportes y estadística



|  |  |
| --- | --- |
| Nombre de caso de uso | Reportes y estadística |
| Propósito | Esta funcionalidad permitirá que el software pueda devolver información sobre su funcionamiento al usuario que la solicite utilizando tablas y gráficos estadísticos. |
| Actores | Personal |
| Actor iniciador | Personal |
| Pre condición | Ninguna |
| Flujo principal | VER ESTADISTICAS   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder ver gráficos estadísticos acerca de los cursos, conferencias y talleres.   VER REPORTES   * Ingresar a su correo y presionar la opción “Crear nuevo mensaje”. * Ingresar como asunto del mensaje el comando respectivo para poder ver reportes acerca del funcionamiento de los cursos, conferencias y talleres. |
| Post condición | Ninguna |
| Excepción | Ninguna |

## 3.4 Modelo general de casos de uso



# Capítulo 3: Análisis

## 4.1 Análisis de la arquitectura

### 4.1.1 Identificación de paquetes

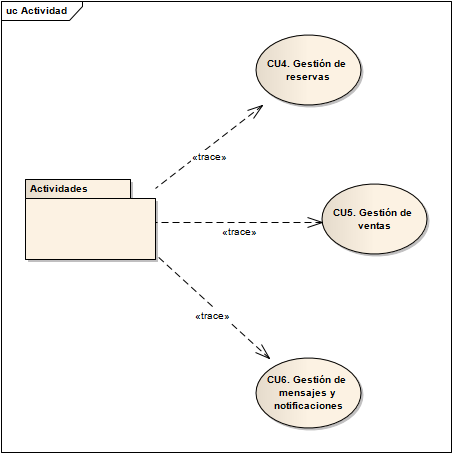
Los paquetes que se definieron son:

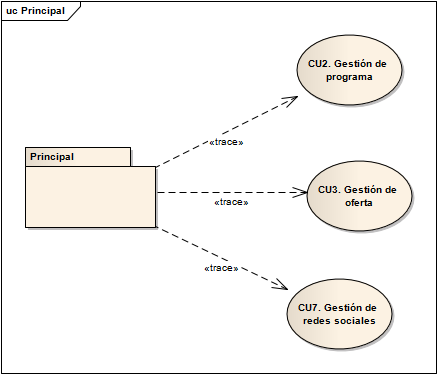
* Administración
* Principal
* Actividades



### 4.1.2 Relación de paquetes y casos de uso

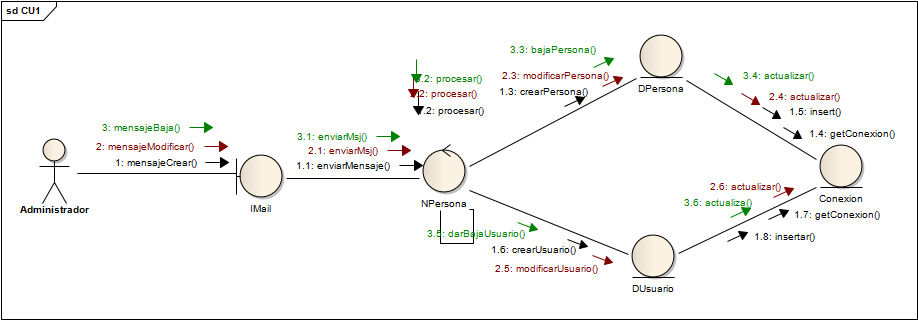




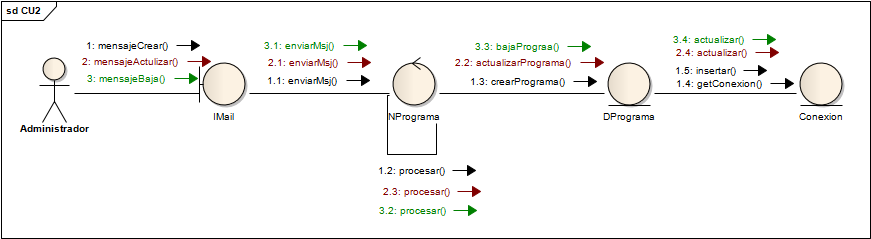


## 4.2 Análisis de casos de uso

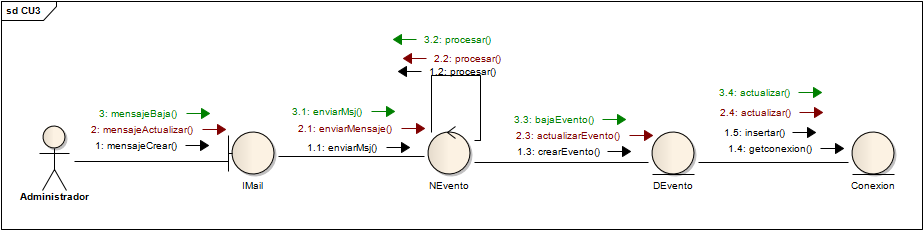
CU1. Gestión de Usuario



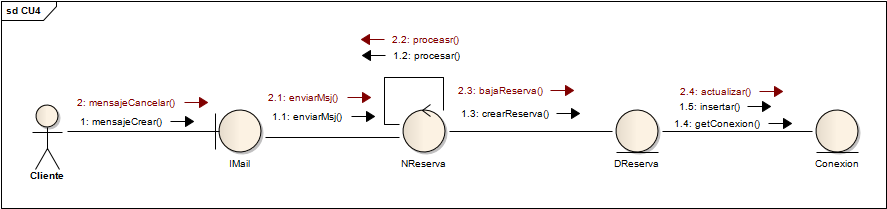
CU2. Gestión de programa



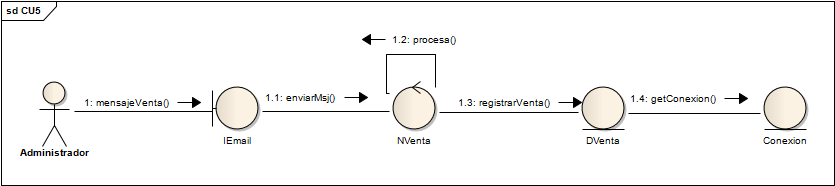
CU3. Gestión de eventos



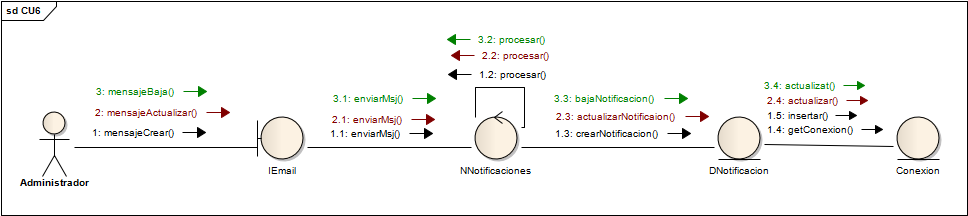
CU4. Gestión de reservas



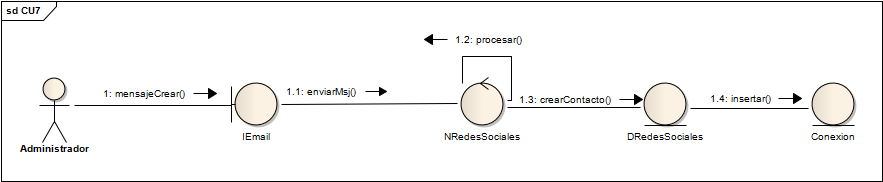
CU5. Gestión de ventas



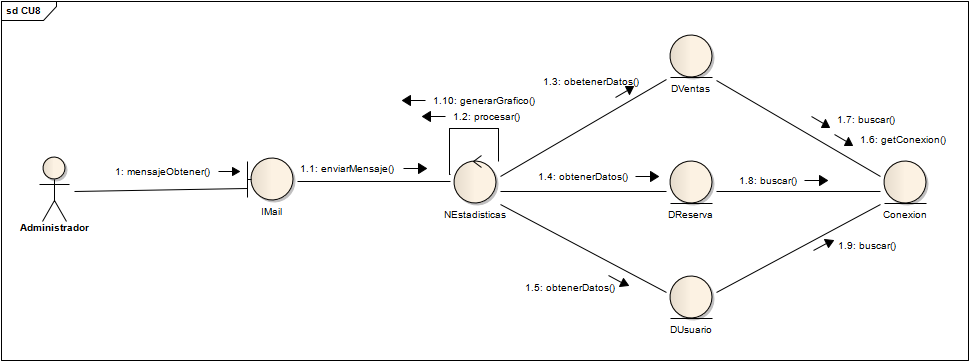
CU6. Gestión de mensajes y notificaciones



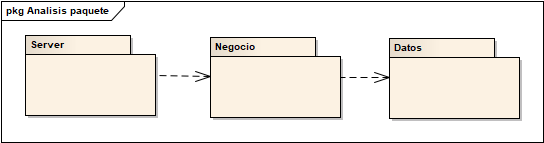
CU7. Gestión de redes sociales



CU8. Reportes y estadística



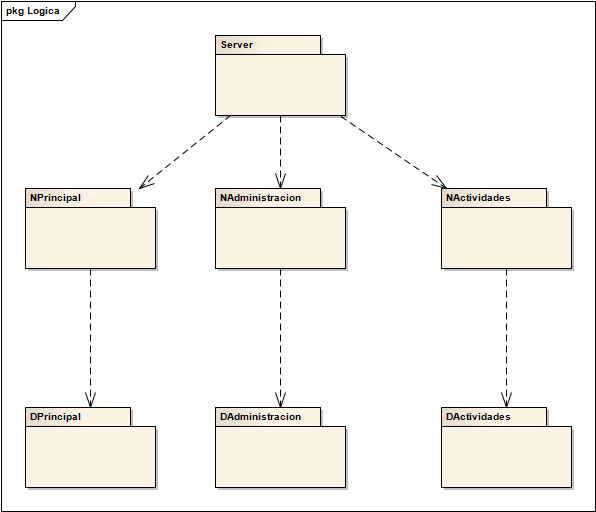
## 4.3 Análisis de paquetes



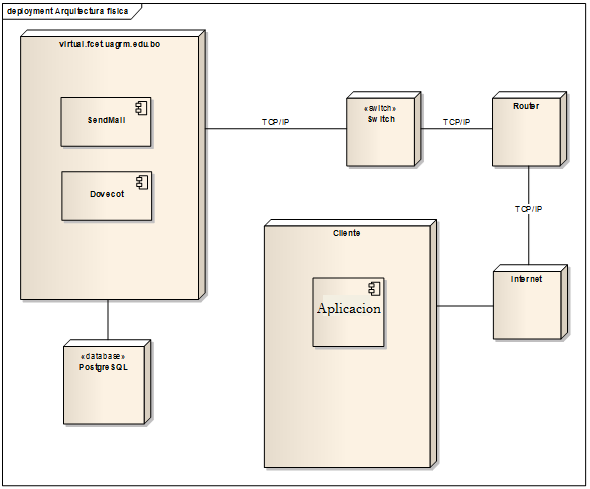
# Capítulo 4: Diseño

## 5.1 Diseño de la arquitectura

### 5.1.1 Arquitectura lógica

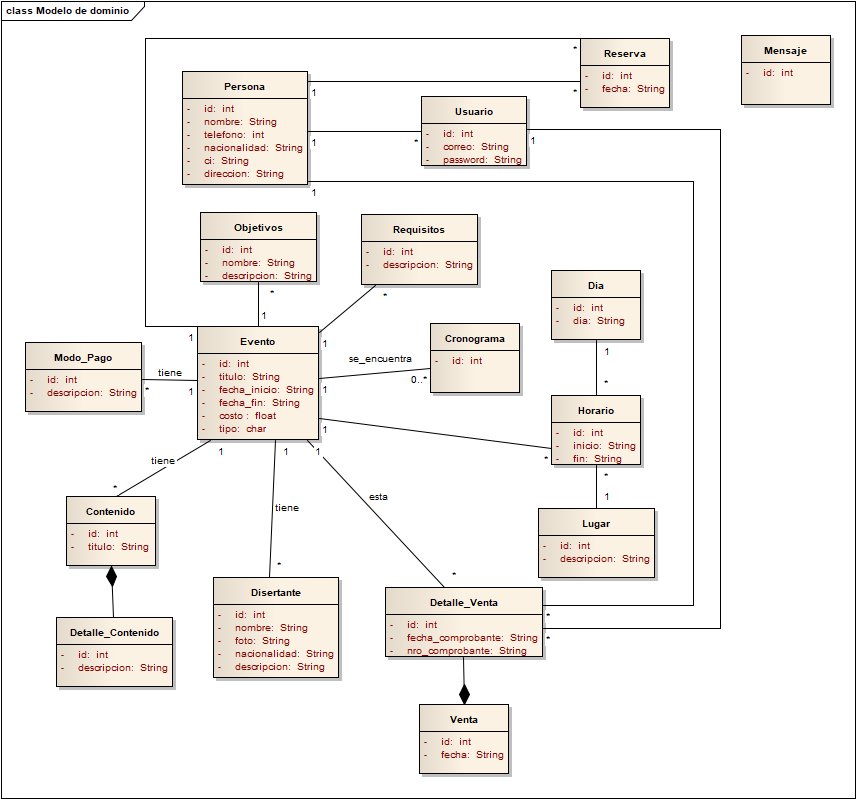


### 5.1.2 Arquitectura física



## 5.2 Diseño de la base de datos

### 5.2.1 Diseño conceptual



### 5.2.2 Diseño lógico

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Persona | | | |  | |  |  |  |  |
| Id | nombre | | | teléfono | | nacionalidad | ci | dirección |  |
| Usuario | | | |  | |  |  |  |  |
| Id | correo | | | password | | **Id\_persona** |  |  |  |
| Evento | | | |  | |  |  |  |  |
| Id | Titulo | | | fecha\_inicio | | fecha\_fin | costo | tipo |  |
| Objetivos | | | |  | |  |  |  |  |
| Id | Nombre | | | descripción | | **Id\_evento** |  |  |  |
| Requisitos | | | |  | |  |  |  |  |
| Id | descripcion | | | **Id\_evento** | |  |  |  |  |
| Modo\_Pago | | | |  | |  |  |  |  |
| Id | descripcion | | | **Id\_evento** | |  |  |  |  |
| Disertante | | | |  | |  |  |  |  |
| Id | Nombre | | | descripcion | **Id\_evento** | |
| Contenido | | | |  | |  |  |  |  |
| Id | Titulo | | | **Id\_evento** | |  |  |  |  |
| Detalle\_contenido | | | |  | |  |  |  |  |
| Id | descripcion | | | **Id\_contenido** | |  |  |  |  |
| Reserva | | | |  | |  |  |  |  |
| Id | | Fecha | | **Id\_persona** | | **Id\_evento** |  |  |  |
| Dia | | | |  | |  |  |  |  |
| Id | | Dia | |  | |  |  |  |  |
| Lugar | | | |  | |  |  |  |  |
| Id | | descripcion | |  | |  |  |  |  |
| Horario | | | |  | |  |  |  |  |
| Id | | | Inicio | Fin | | **Id\_dia** | **Id\_lugar** | **Id\_evento** |  |
| Cronograma | | | |  | |  |  |  |  |
| Id | | | **Id\_evento** |  | |  |  |  |  |
| Venta | | | |  | |  |  |  |  |
| Id | | | Fecha |  | |  |  |  |  |
| Detalle\_Venta | | | |  | |  |  |  |  |
| Id | | | fecha\_  comprobante | nro\_  comprobante | | **Id\_**  **persona** | **Id\_**  **usuario** | **Id\_evento** | **Id\_venta** |
| Mensaje | | |  |  | |  |  |  |  |
| Id | | |  |  | |  |  |  |  |

### 5.2.3 Diseño físico

**Tabla de volumen**

**Persona**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| id | Numero | No | Primaria | No |
| Nombre | Texto | 40 | No | No |
| Teléfono | Numero | No | No | No |
| Nacionalidad | Texto | 25 | No | No |
| Ci | Texto | 10 | No | No |
| Dirección | Texto | 40 | No | Si |

**Usuario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| id | Numero | No | Primaria | No |
| Correo | Texto | 40 | No | No |
| Password | Texto | 40 | No | No |
| Id\_persona | Numero | No | Foránea | No |

**Evento**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| id | Numero | No | Primaria | No |
| Titulo | Texto | 100 | No | No |
| Fecha\_inicio | Texto | 15 | No | No |
| Fecha\_fin | Texto | 15 | No | No |

**Reserva**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| id | Numero | No | Primaria | No |
| Fecha | Texto | 15 | No | No |
| id\_persona | Numero | No | Foránea | No |
| id\_evento | Numero | No | Foránea | No |

**Objetivos**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Nombre | Texto | 40 | No | No |
| Descripción | Texto | 512 | No | No |
| Id\_evento | Numero | No | Foránea | No |

**Requisitos**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Fecha | No | Primaria | No |
| Descripción | Texto | 512 | No | No |
| Id\_evento | Numero | No | Foránea | No |

**Modo\_Pago**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Descripción | Texto | 512 | No | No |
| Id\_evento | Numero | No | Foránea | No |

**Cronograma**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Id\_evento | Numero | No | Foránea | No |

**Dia**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Dia | Texto | 15 | No | No |

**Lugar**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Descripción | Texto | 512 | No | No |

**Horario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Inicio | Texto | 15 | No | No |
| Fin | Texto | 15 | No | No |
| Id\_evento | Numero | No | Foránea | No |
| Id\_dia | Numero | No | Foránea | No |
| Id\_lugar | Numero | No | Foránea | No |

**Contenido**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | No | No |
| Titulo | Texto | 100 | No | No |
| Id\_evento | Numero | 255 | Foránea | No |

**Detalle\_contenido**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Descripción | Texto | 512 | No | No |
| Id\_contenido | Numero | No | Foránea | No |

**Disertante**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |
| Nombre | Texto | 100 | No | No |
| Descripción | Texto | 512 | No | No |
| Id\_evento | Numero | No | Foránea | No |

**Venta**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | No | No |
| Fecha | Texto | 20 | No | No |

**Detalle\_Venta**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | No | No |
| Nro\_  comprobante | Texto | 40 | No | No |
| Fecha\_  comprobante | Texto | 40 | No | No |
| Id\_evento | Numero | No | Foránea | No |
| Id\_usuario | Numero | No | Foránea | No |
| Id\_venta | Numero | No | Foránea | No |

**Mensaje**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo | Tipo de dato | Amplitud | Llave | Nulo |
| Id | Numero | No | Primaria | No |

## Script

CREATE TABLE persona(

id bigserial NOT NULL,

nombre character varying(40) NOT NULL,

telefono integer NOT NULL,

nacionalidad character varying(25) NOT NULL,

ci character varying(8) NOT NULL,

direccion character varying(40),

tipo character varying(2),

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id)

);

CREATE TABLE usuario(

id bigserial NOT NULL,

id\_persona integer NOT NULL,

correo character varying(40) NOT NULL,

password character varying(40) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_persona) REFERENCES persona (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT

);

CREATE TABLE evento(

id bigserial NOT NULL,

titulo character varying(100) NOT NULL,

fecha\_inicio character varying(15) NOT NULL,

fecha\_fin character varying(15) NOT NULL,

costo float NOT NULL,

tipo char(1) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id)

);

CREATE TABLE reserva(

id bigserial NOT NULL,

id\_persona integer NOT NULL,

id\_evento integer NOT NULL,

fecha character varying(15) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_persona) REFERENCES persona (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT

);

CREATE TABLE objetivos(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

nombre character varying(40) NOT NULL,

descripcion character varying(512) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE requisitos(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

descripcion character varying(512) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE modo\_pago(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

descripcion character varying(512) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE cronograma(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

titulo character varying(254),

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE dia(

id bigserial NOT NULL,

dia character varying(15) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id));

CREATE TABLE lugar(

id bigserial NOT NULL,

descripcion character varying(512) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id) );

CREATE TABLE horario(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

id\_dia integer NOT NULL,

id\_lugar integer NOT NULL,

inicio character varying(15) NOT NULL,

fin character varying(15) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

FOREIGN KEY (id\_dia) REFERENCES dia (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

FOREIGN KEY (id\_lugar) REFERENCES lugar (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE contenido(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

titulo character varying(100) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE detalle\_contenido(

id bigserial NOT NULL,

id\_contenido integer NOT NULL,

descripcion character varying(512) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_contenido) REFERENCES contenido (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE disertante(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

nombre character varying(254) NOT NULL,

descripcion character varying(1024) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

CREATE TABLE venta(

id bigserial NOT NULL,

fecha character varying(20) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id) );

CREATE TABLE detalle\_venta(

id bigserial NOT NULL,

id\_evento integer NOT NULL,

id\_usuario integer NOT NULL,

id\_persona integer NOT NULL,

id\_venta integer NOT NULL,

nro\_comprobante character varying(40) NOT NULL,

fecha\_comprobante character varying(40) NOT NULL,

created\_at character varying(14),

updated\_at character varying(14),

deleted\_at character varying(14),

PRIMARY KEY(id),

FOREIGN KEY (id\_evento) REFERENCES evento (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

FOREIGN KEY (id\_usuario) REFERENCES usuario (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

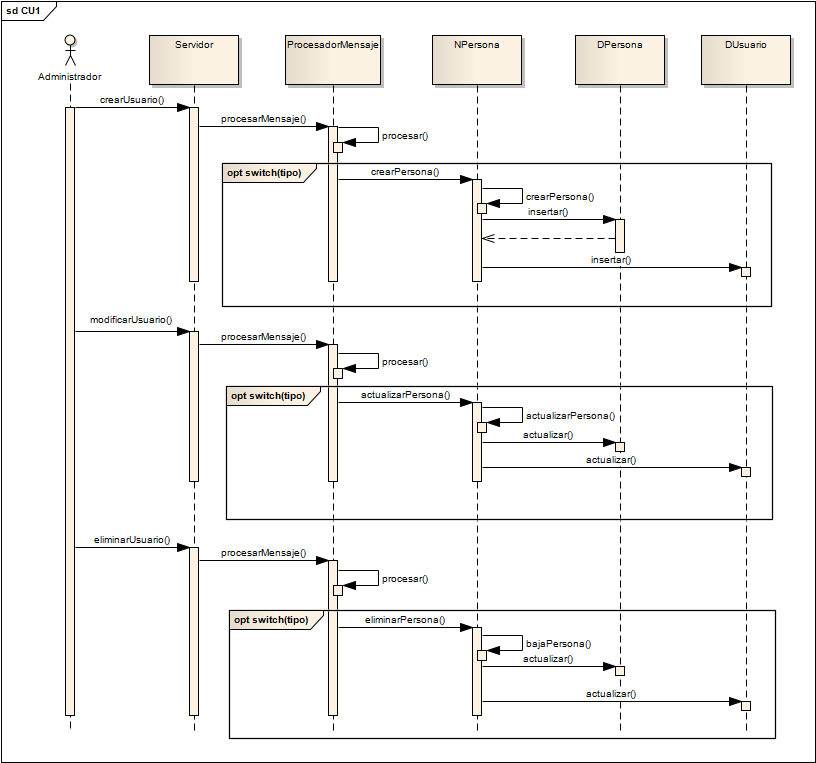
FOREIGN KEY (id\_persona) REFERENCES persona (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT,

FOREIGN KEY (id\_venta) REFERENCES venta (id) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT );

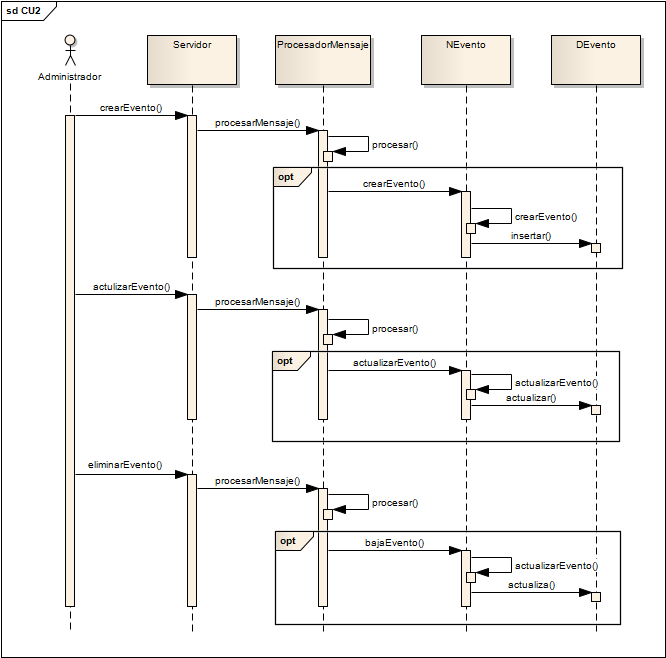
## 5.4 Diseño de casos de uso

### 5.3.1 Diagramas de secuencia

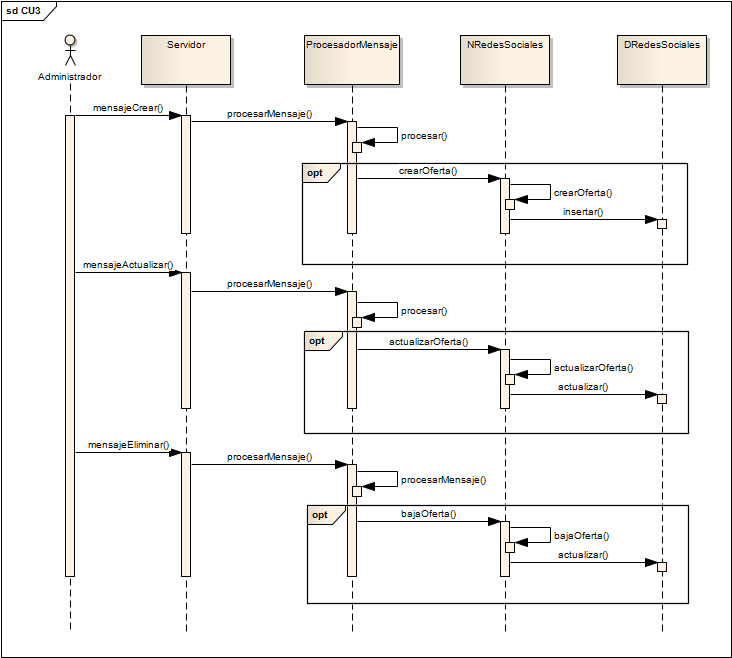
#### CU1. Gestión de usuario



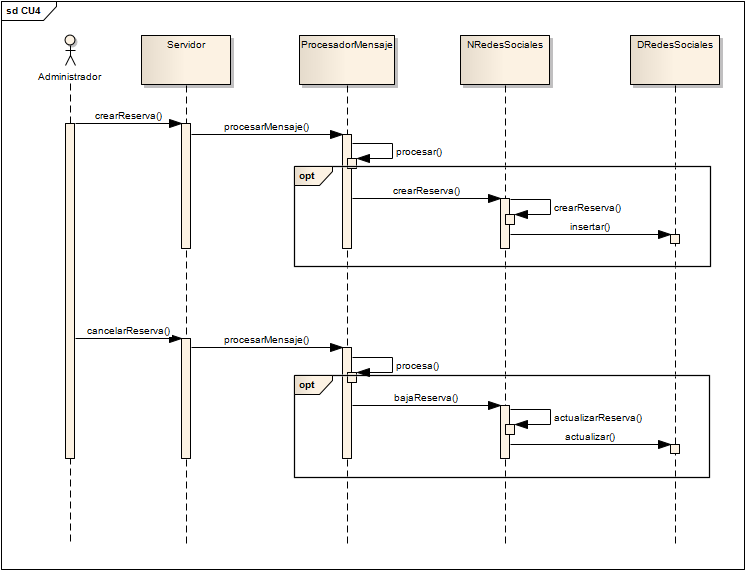
#### CU2. Gestión de programa



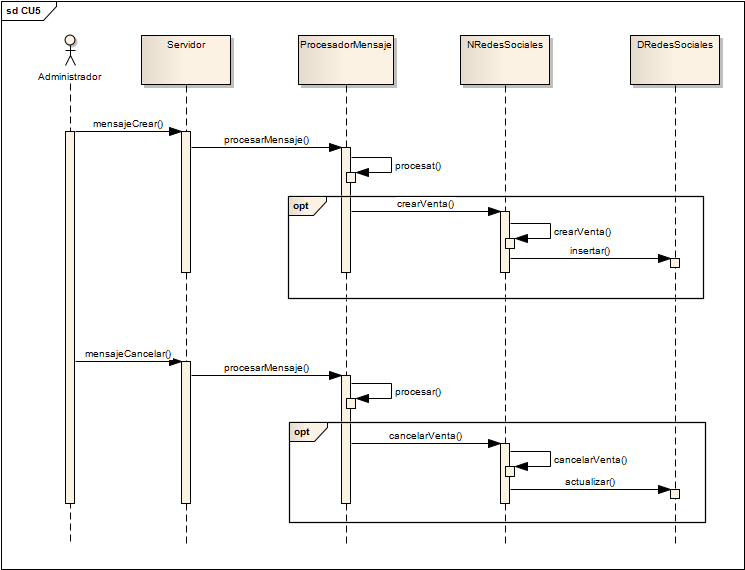
#### CU3. Gestión de oferta



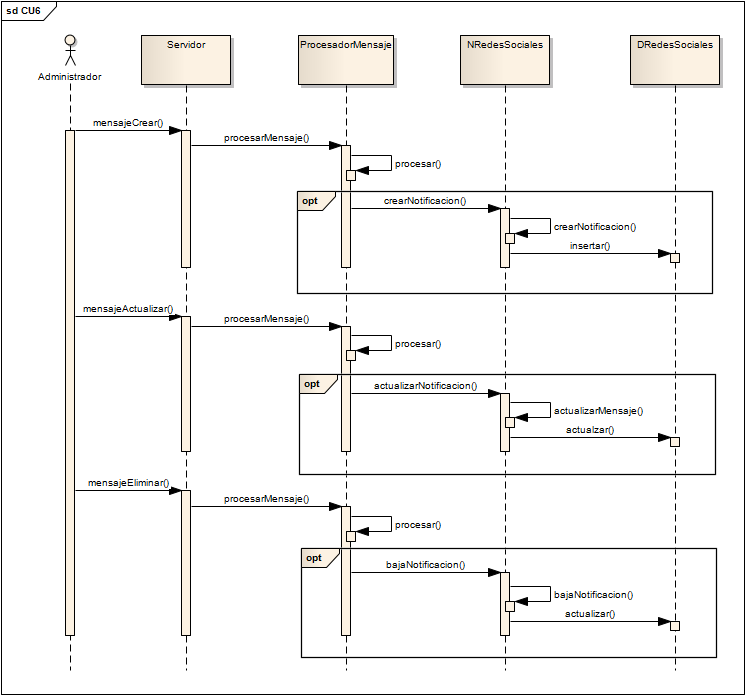
#### CU4. Gestión de reservas



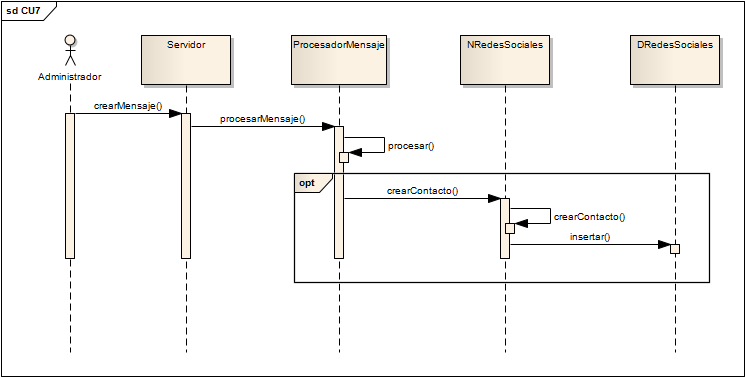
#### CU5. Gestión de ventas



#### CU6. Gestión de mensajes y notificaciones

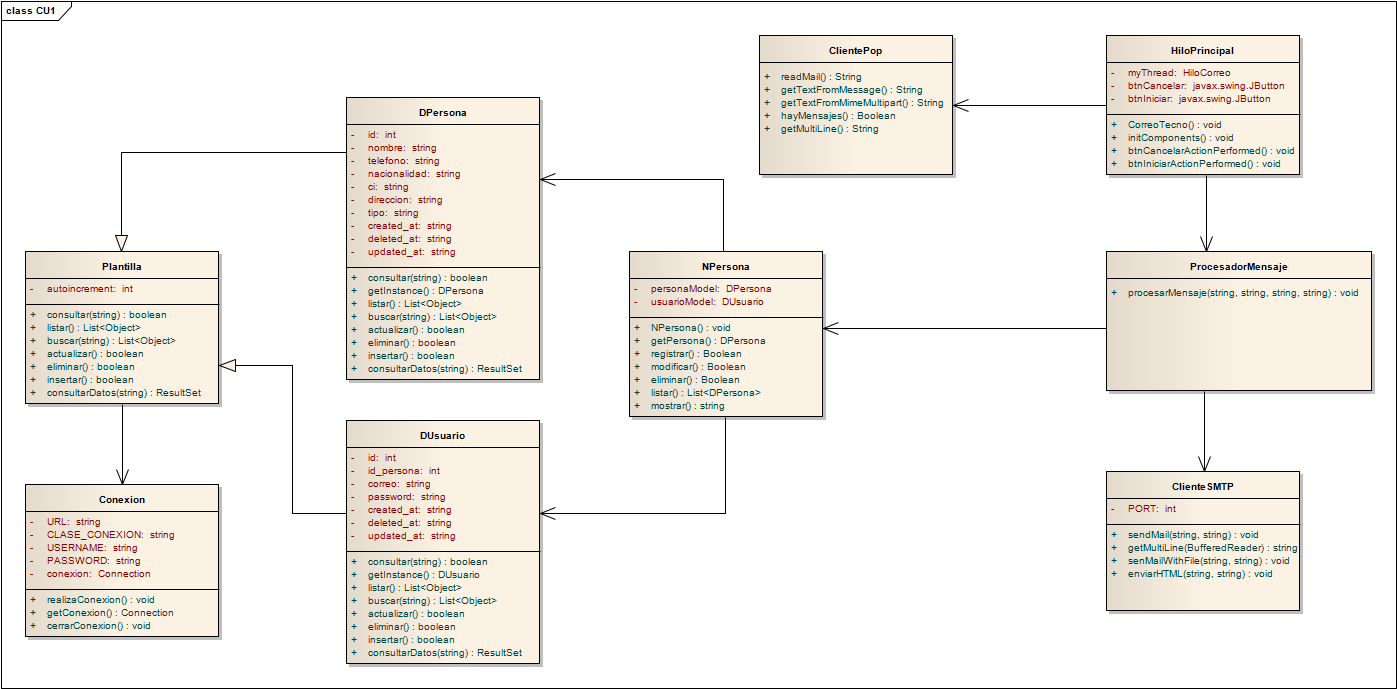


#### CU7. Gestión de redes sociales

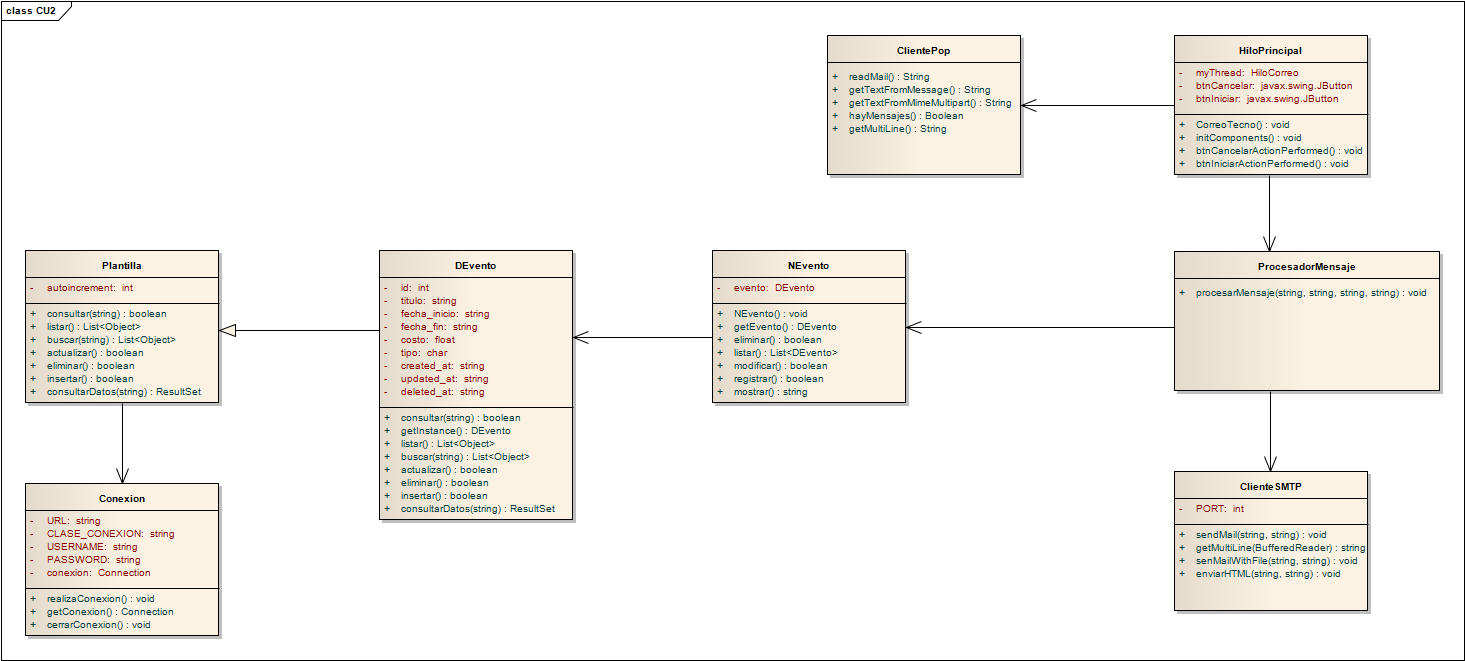


### 5.3.2 Diagramas de detalle procedimental

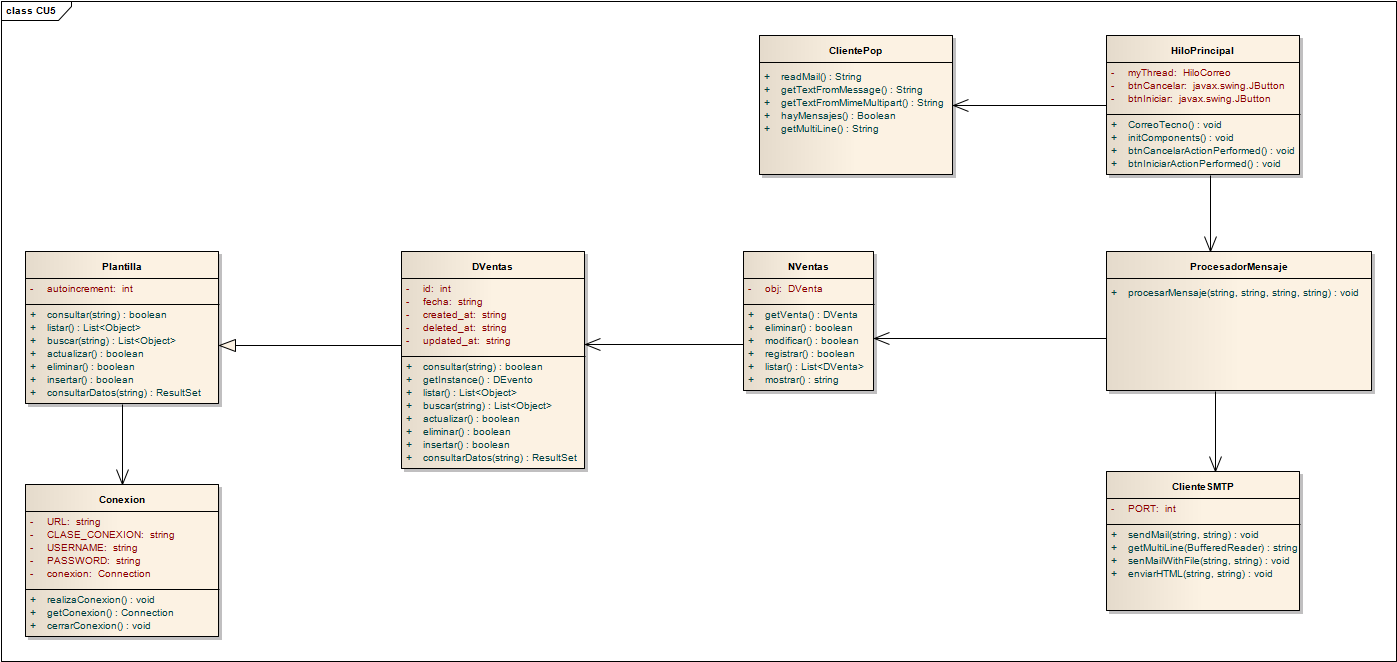
* CU1. Gestión de Usuario

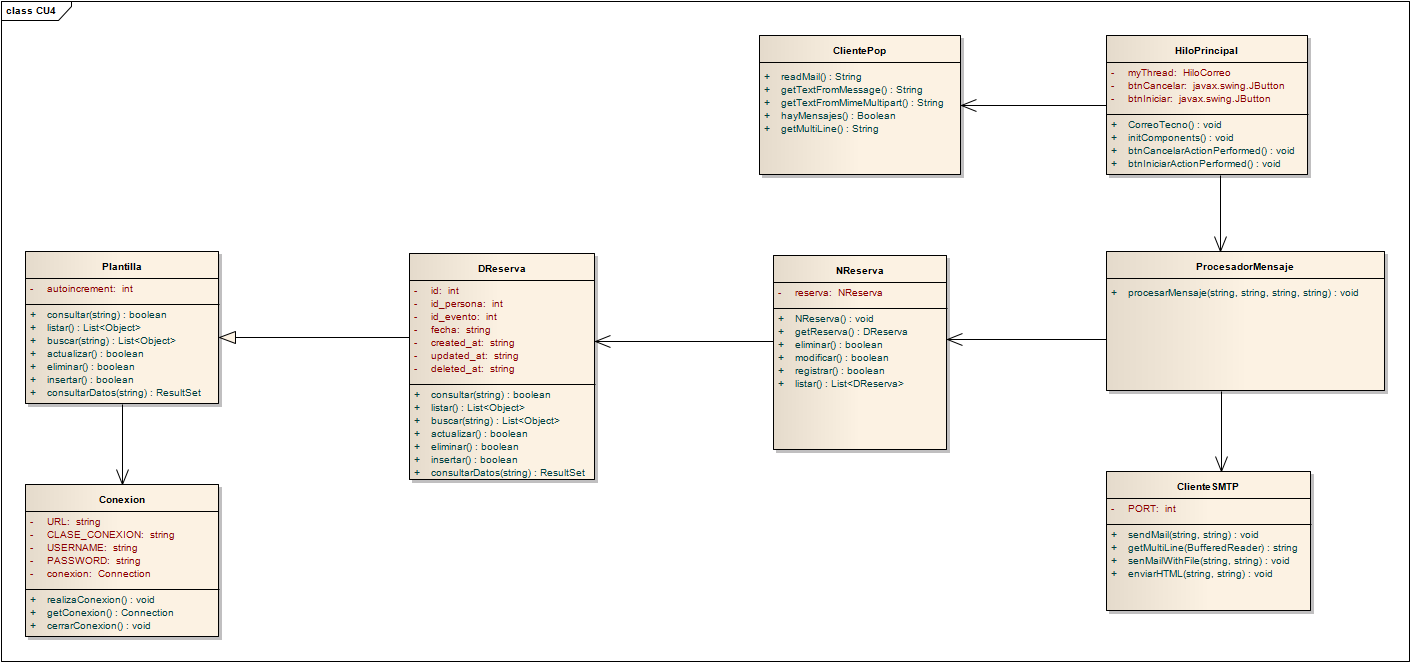


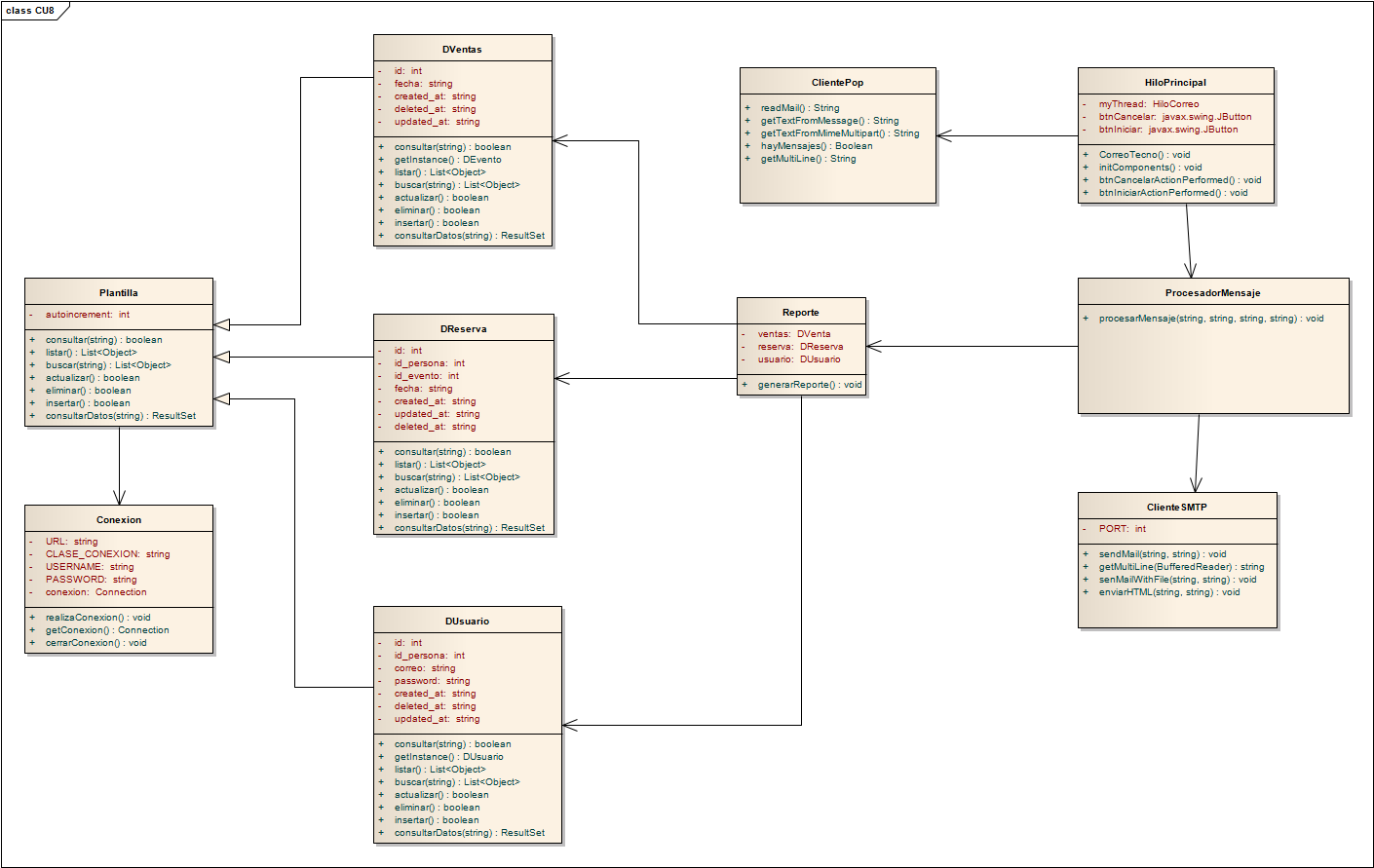
* CU2. Gestión de evento



* CU4. Gestión de reservas



* CU5. Gestión de ventas
* CU8. Reportes y estadística



# Capítulo 6: Implementación

## 6.1 Plataforma de desarrollo de software

### 6.1.1 Lenguaje de programación

**NetBeans Java IDE 8.0.2**

El IDE NetBeans es un entorno de desarrollo integrado (IDE) modular y basado en estándares, escrito con el lenguaje de programación JAVA. El proyecto de NetBeans consta de un IDE de código abierto y gran variedad de funciones; y una plataforma para aplicaciones de cliente enriquecidas que se puede utilizar como marco genérico para crear cualquier tipo de aplicación web. El lenguaje de programación utilizado para el desarrollo del proyecto es el lenguaje JAVA, el cual provee los componentes necesarios para la implementación de los requisitos del proyecto.

### 6.1.2 Base de datos

**PostgreSQL**

PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad e integridad de datos. Se ejecuta en los principales sistemas operativos que existen en la actualidad. Una base de datos de clase empresarial, PostgreSQL cuenta con características avanzadas tales como Multi-Version Control de concurrencia (MVCC), puntos en tiempo de recuperación, tablespaces, replicación asincrónica, transacciones anidadas (savepoints), respaldos online/hot, un sofisticado query planner/optimizer. Soporta el conjunto de caracteres internacional, codificaciones de caracteres multibyte, Unicode, mayúsculas y minúsculas. Es altamente escalable, tanto en la enorme cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede administrar. Hay sistemas activos en PostgreSQL en entornos de producción que manejan más de 4 terabytes de datos. Algunos límites y caracteristicas generales que se incluyen en PostgreSQL son:

|  |  |
| --- | --- |
| Tamaño máximo de la Base de datos | Ilimitado |
| Tamaño máximo de la tablas | 32 TB |
| Tamaño máximo de la fila | 1.6 TB |
| Tamaño máximo para cada campo | 1 GB |
| Máximo de filas por tabla | Ilimitado |
| Maximo de columnas por tabla | 250-1600 dependiendo del tipo de columna |
| Máximo de indices por tabla | Ilimitado |

### 6.1.3 Sistemas operativos

**Sistema operativo Linux Windows 10**

No obstante, nuestro sistema de información soporta cualquier plataforma de desarrollo en el entorno móvil, web y de escritorio.

El lenguaje de programación PHP actualmente está disponible para cualquier sistema operativo de escritorio, tanto para pruebas locales como para desarrollo en el entorno web.

### 6.1.4 Otros

**Architect Enterprise**

Es una herramienta que se utiliza en el desarrollo de software para todo tipo de industrias que permite el diseño UML y la cual abarca por completo el desarrollo de software, desde la administración de requerimientos como el diseño, pruebas y mantenimiento. Esta aplicación incluye los siguientes diagramas: Clases, Objetos, Paquetes, Componentes, Despliegue, Casos de uso, Comunicación, Interacción, Actividad, Secuencia, Tiempo, Navegación, Capas

Máquina de estados.

# Capítulo 7: Pruebas

## CU1. Gestionar usuarios

Se encarga de administrar los datos de los usuarios vía mail

**Entrada:** Se ingresa los datos necesarios del usuario.

Los datos a meterse tienen orden el cual tienen que seguir, este orden es nombre, ci, teléfono, correo, contraseña, sexo después de seguir ese orden se da clic en enviar.

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos y enviar un mensaje de retorno para notificar.

**Condiciones:** El orden de introducción de datos tiene que haber sido respetados

Todos los datos solicitados tienen que haber sido introducidos, no tiene que haber faltado alguno.

**Procedimiento de prueba de registro de usuario**

1. Enviar un mail con el comando de respectivo, bajo el formato especificado como datos de entrada eligiendo el tipo de usuario cliente, personal o fotógrafo.

2. Se recibirá un mail con la confirmación del proceso.

## CU2. Gestionar evento

Se encarga de administrar los datos de los eventos creados por los clientes.

**Entrada:** Se ingresan los datos de los eventos para que sean almacenados en la base de datos

Los datos que se ingresan deben seguir un orden: nombre, codigodeacceso, fechaevento, idzona, idcliente.

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos

**Condiciones:**

* Los datos deben ingresarse en el orden correspondidos y deben ser validos

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente, con los parámetros necesarios.
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

## CU3. Gestion de Oferta

Se encarga de administrar las zonas donde se realizan los diferentes eventos y pedidos.

**Entrada:** Se ingresan los datos pedidos para el formulario para que sean almacenados en la base de datos. Los datos a ingresar tienen un orden el cual tienen que seguir, este orden es: nombre.

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos

**Condiciones:**

* El orden de introducción de datos tiene que haber sido respetados
* Todo el dato solicitado tiene que ser introducidos, no tiene que faltar alguno

**Procedimiento de prueba de registro de zona**

* Enviar un mail con el comando correspondiente, bajo el formato especificado con todos los parámetros de entrada
* Se recibirá un mail con la confirmación del proceso.

## CU4. Gestion de Reserva

Se encarga de administrar los datos de las promociones creadas para los servicios.

**Entrada:** Se ingresan los datos de las promociones para que sean almacenados en la base de datos. Los datos que se ingresan deben seguir un orden: nombre, descuento, fecha inicio, fechafin.

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos

**Condiciones:** Los datos deben ingresarse en el orden correspondidos y deben ser validos

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente, con los parámetros necesarios.
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

## CU5. Gestion Venta

Se encarga de administrar los datos de los servicios que se ofrecen

**Entrada:** Se ingresan los datos de los servicios para que sean almacenados en la base de datos. Los datos que se ingresan deben seguir un orden: nombre, descripción, precio y tipo

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos

**Condiciones:** Los datos deben ingresarse en el orden correspondidos y deben ser validos

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente, con los parámetros necesarios y en orden.
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

## CU6. Gestion de Notificaciones

Se encarga de administrar los diferentes pedidos realizados por los clientes.

**Entrada:** Se ingresan los datos de los pedidos para que sean almacenados en la base de datos

Los datos que se ingresan deben seguir un orden: Direccion, idZona, idEvento, codigoAcceso

**Resultado:** Los datos tienen que haber sido registrados exitosamente en la base de datos

**Condiciones:** Los datos deben ingresarse en el orden correspondidos y deben ser validos

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente, con los parámetros necesarios.
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

## CU7. Gestion de Redes Sociales

Se encarga de administrar los datos de las proformas creadas para los clientes

**Entrada:** Para registrar la proforma no se requiere de ningún parámetro más que utilizar el comando correspondiente.

**Resultado:** La proforma debe registrarse correctamente en la base de datos

**Condiciones:** Solo hacer llamado al comando

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

## CU8. Reportes y estadística

Se encarga de mostrar los reportes y estadísticas sobre la información del estudio fotográfico

**Entrada:** Para mostrar los reportes y gráficos estadístico solo es necesario llamar al método

**Resultado:** Muestra los reportes y gráficos estadísticos con la información acerca del estudio

**Condiciones:** Solo hacer llamado al comando

**Procedimiento de prueba de registro de evento**

* Enviar un mail con el comando correspondiente
* Llegando el mensaje recibirá un mensaje de respuesta notificando si se realizó con éxito o no.

# Bibliografía

Jacobson - booch - rumbaugh/Andrés Otero (2000). Proceso unificado de desarrollo de software (4ta) (3). Madrid-España: ADDYSON WESLEY.

Nicolás Tedeschi (2008). Web Services un ejemplo práctico. Recuperado en: https://msdn.microsoft.com/es-es/library/bb972248.aspx [2018, 16 de mayo].

Evelio Martinez (2011). El correo electrónico y su historia. Recuperado en: http://www.eveliux.com/mx/El-correo-electronico-y-su-historia.html [2018, 16 de mayo].

Sajid Demian Lonngi Reyna (2009). Correo Electrónico. Recuperado en: https://sites.google.com/site/competenciaseinovaciones09/correo-electronico [2018,16 de mayo].

Universidad de Alicante (2016). Protocolos de correo electrónico: SMTP, POP3 e IMAP. Recuperado en: https://moodle2016-17.ua.es/moodle/pluginfile.php/62202/mod\_resource/content/9/comunicacion/page\_05.htm [2018, 17 de mayo].

Netbeans (2017). Bienvenido a NetBeans. Recuperado en: https://netbeans.org/index\_es.html [2018, 17 de mayo].

Yendes Edwin (2015). PostgreSQL. Recuperado en: https://iutablog.wordpress.com/2017/05/11/postgresql/ [2018, 18 de mayo].